A LINEAR ECONOMIC MODEL FOR A NAVAL AIR REWORK FACILITY

William Martin Myers

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A Linear Economic Model
for a Naval Air Rework Facility

by

William Martin Myers

Thesis Advisor: N. K. Womer

September 1972

A Linear Economic Model

for a Naval Air Rework Facility

by

William Martin Myers
Lieutenant, Supply Corps, United States Navy
B.S., North Carolina State University, 1964

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1972

ABSTRACT

A prototype management information system called Work-in-Process Inventory Control System (WIPICS) has been installed at the Naval Air Rework Facility (NARF), North Island. WIPICS is expected to benefit the NARF by reducing rework times and rework costs. By assuming that linear production processes may be estimated from statistical data accumulated by the NARF, a linear economic model is constructed which predicts a required budget in man-hours, material cost and overhead cost categories for a specified production output level. Sample problems are solved and parametric studies are done to determine changes in the required budget for restrictions in man-hours, material cost and overhead cost. The model will be used as an aid in the cost-effectiveness evaluation of WIPICS by the Management Systems Development Office.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENT

The author wishes to thank Assistant Professor James K. Hartman of the Naval Postgraduate School for contributing the computer programs appearing in Appendixes C, D and E.

I. INTRODUCTION

The Naval Air Rework Facility (NARF) North Island is currently in the process of installing and evaluating a prototype of a recently developed management information system. The system is called the Work-in-Process Inventory Control System (WIPICS) and was designed and developed by the ROHR Corporation, Chula Vista, California. The system employs the latest advances in computer technology which include real time data file updating and information retrieval with communications via a "touch-tone" telephone system and remote teletypewriter terminals. WIPICS contains data regarding individual items and their relationship to the next higher or lower assembly. The system provides an immediate audio (pre-recorded voice) response for simple answers and printed reports for more detailed answers. Each individual item or component has a unique identification number (register number) which is assigned and used for all WIPICS transactions. As an item changes location or status a transaction immediately updates the computer storage record and any subsequent queries will be answered with current information.

The WIPICS formulation concept began in early 1968 when RADM. J. Smith (then NAVAIRSYSCOMREPAC) conducted a review of then current management information systems. RADM Smith was convinced that the Uniform Automatic Data Processing

System (UADPS) then being implemented at all Industrial Naval Air Stations (INAS) was rapidly being outdated by the tremendous advances being made in computer technology. Consequently NAVAIRSYSCOM activities were no longer using the most modern management systems available. A "task force" was created and the resulting study was a systems design proposal for the establishment of the Management Systems Development Office (MSDO).

In early 1969 MSDO and the ROHR Corporation studied NARF North Island to determine the applicability of an advanced management information system currently being used by ROHR. The study conclusions led to the development of a contract request for WIPICS by MSDO in late 1969. By July 1970 approval had been granted by SECNAV for MSDO to contract for the construction of a prototype WIPICS to be installed at NARF North Island. The contract negotiation was completed in January 1971. The WIPICS prototype is now installed at NARF North Island and is in an evaluation stage. The expected benefits of WIPICS are three-fold:

(1) Allow positive control of all work-in-process

(2) Reduction of turn-around time for repairs

(3) Improved production rates

Other benefits would be possible if extensions are made to WIPICS:

(1) Attendance and labor force data

(2) Association of tooling and technical data

(3) Integrate directly with automated storage areas.

In addition to the approval of a WIPICS prototype in July 1970, SECNAV placed a requirement on MSDO to evaluate the benefits of WIPICS in accordance with the Resources Conservation Program. The documentation for this cost justification was to be developed by MSDO with the methods and data subsequently audited by the Naval Audit Service. Using data collected and summarized by MSDO, the Operations Research/Administrative Sciences Department of the Naval Postgraduate School agreed to conduct a study to develop a methodology for quantitative cost-effectiveness reviews for technological changes of this type.

A brief comparison of the UADPS Feedback system and WIPICS is presented in Figure 1.

B. STATEMENT OF THE PROBLEM

The overall project being considered by the Naval Postgraduate School is the development of a methodology for quantitative cost-effectiveness analysis of technological changes at a particular organization. The results will be specifically applicable to the evaluation of WIPICS at NARF North Island, but may be generalized to any organization which produces several products and incorporates a technological change either in a production process or the control of a production process. The results of this cost-effectiveness analysis will determine the future of WIPICS as a management tool for possible use at other INAS as well as continued use at NARF North Island.

	UADPS Feedback	WIPICS
Data Age/Value	8 to 32 hours from time of occurrence (minimum)(history)	Seconds after occurrence (current)
Data Elements	Comprehensive ID and status for individual and end items	Minimum ID, status for individual and end items and <u>clear</u> <u>inter-relation</u> of items to one another
Data Response	Voluminous printed reports for a general area or program available to limited audience	<u>Specific</u> response available throughout the plant to all need to know
Item Relationship	Difficult to recognize sub-items of a higher or lower assembly or operation	Items are chained to their next higher or lower assembly by register number
Process	Batch sequential - must go from end to end of file on a cyclic basis (minimum daily)	Direct access - go to specific record on file in random order on as needed basis

Figure 1. Comparison of UADPS Feedback and WIPICS

The general approach to the development of a methodology will be to estimate production functions of all major programs at NARF North Island both before and after WIPICS installation. A production function involves determining a specific relationship between a selected output measure and selected input variables. By estimating production functions for entire programs it is possible to avoid the measurement of precise relationships between WIPICS and the NARF North Island shops. Two different types of production functions are to be estimated:

(1) The Cobb-Douglas Production function

(2) A Constant Elasticity Substitution Production function.

To formulate continuous production functions of this type requires a substantial amount of raw data aggregation and sophisticated regression analysis techniques. In order to validate the models using the above techniques a third model will be constructed. This model will be based on Linear Programming Techniques and will yield a linear approximation to the production functions.

Once the before and after WIPICS production functions have been estimated, the results will be analyzed to determine the effects of WIPICS. The WIPICS effects on production efficiency will be translated into dollar savings for comparison with WIPICS costs.

C. SCOPE OF THIS THESIS

Linear Programming Techniques will be used to formulate two Linear Economic Models for production at NARF North Island. The two models will be for the engine program and the aircraft program. Both models will be examined in detail by using sample production requirements and conducting parametric studies to determine the sensitivity of the solutions obtained. No attempt will be made to compare these models with the continuous models (which are being derived separately) in this thesis.

II. DATA ANALYSIS AND REDUCTION

A. DATA AVAILABLE

The Naval Air Rework Facility North Island and the Management Systems Development Office provided actual historical data for use in constructing and testing the linear economic model. The data is collected by the NARF for submission of reports to higher authority and for budgetary/accounting purposes. For each engine or aircraft that is reworked by NARF North Island certain statistical information is accumulated:

(1)	Type of engine or aircraft	
(2)	Identification number	
(3)	Type of work done	
(4)	Induction date	
(5)	Production date	
(6)	Production load norm (man-hours)	(NORM)
(7)	Airframe change man-hours	
(8)	Direct labor hours expended	(DMHR)
(9)	Direct labor cost	(DLB$)
(10)	Direct material cost	(DML$)
(11)	Applied overhead cost	(DOH$)
(12)	Navy Industrial Fund (NIF) rate	(NIFR)

The manner in which the above statistics are accumulated and used by NARF North Island are enumerated in Ref. 10.

The MSDO personnel furnished a summary listing of the above statistical information for each engine and aircraft reworked at the NARF for the period February 1970 through July 1971. The basic data consisted of approximately 1500 observations on the engine program and 360 observations on the aircraft program. Each "observation" consists of a complete record of the above statistical data for a completed NARF job. This data is listed for reference purposes in Appendix A of Ref. 11. In order to keypunch the basic data information it was necessary to assign codes for different types of engines/aircraft and the types of work done. These codes are listed in Tables I and II.

B. SELECTION OF INPUT/OUTPUT VARIABLES

From the basic data it was necessary to select input/output variables to be used for the linear economic model. A single variable consisting of NORM was selected for an output measure.

Input variables were selected to be direct man hours expended (DMHR) direct material cost (DML$), applied overhead cost (DOH$) and number of days in shop (NDAY). The number of days in shop was defined to be production date minus induction date. These input variables were selected because they were thought to represent the variable costs of production.

The expected benefits of WIPICS as stated in Chapter I would be expected to cause measurable decreases in the

Table I

Codes Assigned to the Engine Program

Engine Type	Code	Engine Type	Code
T58-GE-1 (A/F)	51	T64-GE-3 (A/F)	63
T58-GE-3 (A/F)	53	T64-GE-6B	65
T58-GE-5 (A/F)	55	T64-GE-6B (PAWN)	66
T56-A-8P	56	T64-GE-7 (A/F)	67
T58-GE-5 (C/G)	57	T64-GE-413	69
J57-P-4A/22	71	T58-GE-8F	81
J57-P-10	72	T58-GE-8B	82
J57-P-20A	73	T58-GE-8B/F	83
J57-P-22	74	T58-GE-8B (C/G)	84
J57-P-420	75	T58-GE-8B (HH2)	85
		T58-GE-8B/F (CONV)	86
J79-GE-8B	91	T58-GE-10	89
J79-GE-8B/C	92		
J79-GE-8B (RDTE)	93		
J79-GE-10	95		

Work Type *	Code
OVHL	01
OVHL/CONV	02
PAR/REP	03
SUP/REP	04
SUP/REP/CONV	05

* Abbreviations are defined as follows:

OVHL = overhaul
CONV = conversion
PAR = planned aircraft repair
SUP = supply
REP = repair
SEA = South East Asia
CONUS = Continental United States

Table II

Codes Assigned to the Aircraft Program

Aircraft Type	Code
C-2A	10
E-2A/B	11
F-4J	21
F-4B	22
F-4G/B	23
F-8J	25
F-8H	26
RF-8G	27

Aircraft Type	Code
CH-3B	31
RH-3A	32
SH-3A	33
SH-3A/G	34
SH-3D	35
CH-46A	41
CH-46D	42
CH-46F	43
UH-46A	44
UH-46D	45
CH-53A	48
CH-53D	49

Work Type**	Code
OVHL	1*
PAR	2*
PAR/CONV	3*
PAR/MOD	4*
PAR/MOD/REP	5*
PAR/REP	6*
PAR/SEA	7*
PAR/CONUS	8*

* Second digit refers to the cycle number of the aircraft. (A zero means not applicable.)

** See Table I for Work Type abbreviations.

input variables per unit of output demanded. These decreases can in turn be associated with dollar savings and therefore used as a measure of effectiveness for WIPICS.

As a first step in determining the relationships between the available raw data elements, a computer program was written to compute the correlation coefficients between each pair of variables. The results are shown as Tables III and IV. The computer program is described in Appendix A.

C. CLUSTER ANALYSIS

To formulate a linear economic model it is necessary to define a process. A process consists of a mathematical description of the amount of resources used (i.e. labor, material, etc.) to produce a unit amount of an output measure. The problem is how are the raw data observations used to estimate a process. The most natural way would be to estimate a process as an average amount of resources required to do a specific type of work on a certain engine or aircraft. This procedure may not be desirable because there may be alternate processes available to produce the same output. For example, to overhaul an F-8J it may be possible to use a "normal" amount of man-hours and have a low material cost. An alternate process may exist whereby the same work may be accomplished by spending <u>more</u> money on material and <u>reducing</u> the man-hours required.

In an attempt to identify these alternate methods within the basic data observations a clustering algorithm was

CORRELATION FOR RAW DATA - ENGINES

	NORM	DMHR	DLB$	DML$	DOH$	NIFR	NDAY
NORM	1.00000	0.94832	0.94945	0.67620	0.93864	-0.04526	0.47849
DMHR	0.94832	1.00000	0.99695	0.65128	0.99032	-0.10610	0.54876
DLB$	0.94945	0.99695	1.00000	0.65169	0.98909	-0.10374	0.54283
DML$	0.67620	0.65128	0.65169	1.00000	0.65288	0.54323	0.45987
DOH$	0.93864	0.99032	0.98909	0.65288	1.00000	-0.09700	0.55224
NIFR	-0.04526	-0.10610	-0.10374	0.54323	-0.09700	1.00000	0.04808
NDAY	0.47849	0.54876	0.54283	0.45987	0.55224	0.04808	1.00000

	MEAN	STD DEV
NORM	534.25	367.59
DMHR	534.12	376.91
DLB$	3230.61	2258.22
DML$	5764.03	5396.77
DOH$	4045.30	2911.78
NIFR	2515.00	1021.30
NDAY	33.06	10.61

Table III. Correlation Coefficients for Engine Raw Data

CORRELATION FOR RAW DATA - AIRCRAFT

	NORM	DMHR	DLB$	DML$	DOH$	NIFR	NDAY
NORM	1.00000	0.97123	0.96585	0.60199	0.95801	-0.32822	0.86055
DMHR	0.97123	1.00000	0.99633	0.62193	0.97562	-0.36636	0.87527
DLB$	0.96585	0.99633	1.00000	0.61766	0.96923	-0.36199	0.87298
DML$	0.60199	0.62193	0.61766	1.00000	0.61041	0.38659	0.57975
DOH$	0.95801	0.97562	0.96923	0.61041	1.00000	-0.32157	0.84932
NIFR	-0.32822	-0.36636	-0.36199	0.38659	-0.32157	1.00000	-0.29888
NDAY	0.86055	0.87527	0.87298	0.57975	0.84932	-0.29888	1.00000

	MEAN	STD DEV
NORM	8357.50	4043.53
DMHR	8558.26	4385.91
DLB$	52886.70	27281.99
DML$	21155.59	11208.13
DOH$	54912.00	27878.43
NIFR	1529.10	132.64
NDAY	83.58	34.39

Table IV. Correlation Coefficients for Aircraft Raw Data

developed based on a method described in Ref. 8. A series of four computer programs are necessary to achieve the final results. These programs and the purpose of each are described in Appendices B through E. The general procedure is to select a group of observations containing several different types of engines (or aircraft) and some different types of work within each engine group. These observations are then analyzed mathematically for similarity of a selected vector of input/output measures. Similarity among observations is defined as those vectors that use input resources in approximately the same fixed proportion for a unit output measure.

Several trial selections were made with both engine and aircraft data. The engine data was observed to cluster groups of the same engine type together with a high degree of consistency. For example if 20 J-79 engines were selected as part of a trial run of 100 observations, a typical result would be for 17 or 18 of the original 20 to be clustered in the same group. However, the trial selection with the aircraft data were not nearly so consistent. The cluster analysis showed the aircraft data had wide variations among the input variables. The clusterings observed were highly irregular and containing a mixture of different aircraft and different work.

Based on the results of the engine cluster analysis a decision was made to form processes consisting of only

similar engines (or similar aircraft) with identical types of work.

D. DATA AGGREGATION

For the engine data the identification number consisted of the job order number used at the NARF. The job order number consists of a series of alphameric codes as specified in Ref. 10. Using these codes the entire engine raw data deck was sorted into job order number. Similar engine/work types were then grouped together within each job order. The resulting raw data deck was then ordered by engine type, job type and calender quarter in which the work was done (in special cases where a very few observations were available the engine/job types were not separated into calender quarters). Each calender quarter then represented a separate "process observation" on the NARF. At this point the data deck showed 47 different engine/work types and 103 processes by which the work could be done. A computer program was used to examine the dominance relationships among all processes with the <u>same</u> engine/work type. This computer program is described in Appendix G. The dominance relationships among processes are very important in a linear program. For example, if process 1 and process 2 produce the same output but process 2 uses <u>more</u> of every required resource than process 1, then process 1 dominates process 2. In a linear program that minimizes costs, process 1 will always be selected as preferred to

process 2. Therefore, if the linear programming model is to have alternate processes that reflect choices to be made then the dominance must be eliminated between all processes with the same output.

The computer program in Appendix G was run with the 103 processes as previously determined based on job order numbers. The dominance relationship between each pair of processes that produced the same output was examined. If no dominance existed then that group of alternate processes was left alone. If dominance existed among any of the processes in the group then a subjective decision was made to combine two of the processes. For example, if there were five processes and dominance existed between three pairs of them, then two processes would be selected to be combined. The decision on which pair to combine was entirely subjective. The group would then have four alternate processes remaining. The dominance computer program would then be run again and the remaining alternate processes would be examined for dominance. This procedure was repeated until <u>all</u> dominance was eliminated within process groups that had the <u>same</u> output. Since the outputs were intended to be measured separately (as opposed to an aggregate output measure) it was not necessary to examine the dominance between processes of <u>different</u> outputs.

In combining individual raw data observations into estimated processes an aggregation computer program was used. This program is listed in Appendix F. The program

estimates the process by calculating an average value for each variable on the raw data cards. The resulting engine raw data was aggregated from 103 processes to 81 processes in order to eliminate all "within" group dominance.

The aircraft data consisted of 365 raw data observations. Each type of aircraft and type of work considered as a separate output meant that 70 outputs were required. This small amount of data then had to be used to estimate not less than 70 distinct processes. A decision was made to estimate only a single process for each output because of the limited data available. In this case there were no alternate processes so it was not necessary to check for dominance. The linear program in this case would not have a choice among processes and therefore the solution becomes trivial. Further discussion of the aircraft data is contained in Chapters IV and V.

The final listings of the estimated processes for the aircraft and engine programs are included as Appendixes H and I respectively.

III. LINEAR ECONOMIC MODELS

A. INTRODUCTION

Linear programming is a mathematical technique for solving constrained optimization problems of a special type. A linear economic model is a specific type of linear program consisting of the maximization (or minimization) of a linear function of n variables subject to m linear inequality constraints. Most unconstrained optimization problems may be solved by the determination of critical points and their nature (maximum, minimum or saddle point) through solution of first and second derivative equations. In the case of a linear function the first derivative is a non-zero constant and all higher order derivatives are zero. The addition of a constraint set to the linear function further complicates the mathematics. However, several mathematical tools have been developed to solve optimization problems where the objective function and the constraints are linear functions. The first explicit use of linear programming techniques occurred in connection with planning activities for the U.S. Air Force in the late 1940's. Linear programs (and their close mathematical relative the input-output models) have become increasingly important to microeconomic theory in recent years.

In the linear economic model being constructed it is necessary to define a process. A linear production process

is an activity by which one or more outputs are produced in fixed proportions by the application of one or more inputs in fixed proportions. The production process as defined is homogeneous of degree one which implies constant returns to scale. For example, if all inputs to a process are doubled then the output will also be doubled. A linear production function is formed from a collection of linear production processes that may be used simultaneously.

The optimal solution to a linear economic model as described above consists of finding the combination of m processes from n available processes such that a linear objective function is maximized (or minimized). Detailed information concerning linear programming and microeconomic theory can be found in Refs. 1 through 5 and in Ref. 9.

B. ASSUMPTIONS

The following assumptions are made in the formulation of a linear economic model:

(1) The estimated processes (reworking of engines and aircraft) are linear functions and therefore these processes exhibit constant returns to scale.

(2) The above linear processes may be estimated by the aggregation of a finite set of observations over some time period.

(3) The NARF is not a profit maximizing organization, therefore the management objective of the NARF will be assumed to be minimization of costs subject to completion of all work demanded by the operational forces of the Navy.

(4) Prices used in the model are constant and may be estimated from the production data furnished by the NARF.

C. FORMULATION OF THE LINEAR ECONOMIC MODEL

To present the model it is necessary to have some precise mathematical notation available. Definitions will be as follows:

m = number of output constraints

n = number of production processes

($n \geq m$)

R = column vector of available resources $= \begin{bmatrix} R1 \\ R2 \\ R3 \end{bmatrix}$

where

$R1$ = resource for direct man-hours expended per man-hour of NORM

$R2$ = resource for direct material cost in dollars per man-hour of NORM

$R3$ = resource for direct overhead cost in dollars per man-hour of NORM

P = column vector of prices associated with R1, R2 and R3 respectively $= \begin{bmatrix} p \\ 1.0 \\ 1.0 \end{bmatrix}$

where p = price of labor in dollars per man-hour

Z = column vector of activity levels $= \begin{bmatrix} z_i \end{bmatrix}$

$i = 1,\ldots,n$

T = technology matrix of observed processes at NARF [$(m+3) \times n$ matrix] $= \begin{bmatrix} T_1 \\ \hline T_2 \end{bmatrix}$

where

$$T_1 = \text{units of output (NORM)} \quad (m \times n \text{ matrix}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 1 & 0 & & \cdots & 0 \\ 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & & & & & & \vdots \\ 0 & \cdot & \cdot & \cdot & \cdot & 0 & 1 \end{bmatrix}$$

and

$$T_2 = \text{resource matrix} = \left[r_{ij}\right]$$

$$i = 1, 2, 3$$

$$j = 1, \ldots, n$$

with r_{ij} = amount of $i^{\underline{th}}$ resource used per unit output of $j^{\underline{th}}$ activity.

Y = column vector of production output desired in man-hours of NORM. $= \left[y_i\right]$

$$i = 1, \ldots, m.$$

$$T_3 = \text{diagonal matrix of number of days in shop per unit of output} = \begin{bmatrix} t_1 & & & & 0 \\ & \ddots & & & \\ & & t_i & & \\ & & & \ddots & \\ 0 & & & & t_n \end{bmatrix}$$

C = column vector of penalty costs $= \left[c_i\right]$

$$i = 1, \ldots, n$$

where c_i = penalty cost in dollars incurred for each day in shop associated with the $i^{\underline{th}}$ process.

Now that a basic mathematical vocabulary exists for the variables, processes and costs, the linear economic model will be presented and discussed:

objective function minimize $P^T T_2 Z + C^T T_3 Z$

constraints

subject to: $$\begin{bmatrix} -T_1 \\ \hline T_2 \end{bmatrix} \cdot \begin{bmatrix} Z \end{bmatrix} \leq \begin{bmatrix} -Y \\ \hline R \end{bmatrix}$$

The objective function consists of two terms:

$P^T \cdot T_2 \cdot Z$ = actual cost in dollars of resources used for activity level vector Z

$C^T \cdot T_3 \cdot Z$ = penalty cost in dollars for activity level vector Z.

The constraint set may also be separated into two parts for discussion:

$-T_1 \cdot Z \leq -Y$ is a set consisting of the first m rows of the constraints which force the activity level vector Z to choose a set of processes which satisfy the production vector Y.

$T_2 \cdot Z \leq R$ consists of the last 3 rows of the constraint set and ensures that actual resources used do not exceed available resources.

D. SOLUTIONS OF THE MODEL

It is clear that the size of the resource vector R is highly dependent on the production vector Y. In fact these two vectors must be chosen carefully to avoid an infeasible

linear program. The easiest approach to avoiding an infeasible linear program is to make the resource vector R very large, therefore the optimal solution to the model will not be constrained by an active resource constraint. In economic terms this means that the "customer" is willing to pay the NARF as much as necessary to accomplish the work specified by production vector Y. If the NARF is assumed to be efficiently operated, then the costs incurred will be minimized.

In summary, the objective function of the model will be a minimum when the model is not constrained by the resource vector R. In other words, the costs are minimum when the resource constraints are inactive constraints. The value of the objective function represents the total cost incurred to accomplish production vector Y based on past performance data of the NARF.

It should be noted that the linear economic model presented in this chapter is not a production management tool in the sense that the model chooses "processes" by which the NARF should rework aircraft or engines. The model only provides a budget cost plus a penalty cost for a specified amount of work to be done. Some information is available from the model concerning tradeoff values among the three resources. The model can also determine the minimum amount of any single resource required to accomplish the specified work. Some example solutions are presented and discussed in the next chapter.

IV. SAMPLE PROBLEMS

A. INTRODUCTION

Two separate models were constructed using forecasted workload requirements during third quarter, fiscal year 1972. The forecasted requirements for the engine and aircraft programs were obtained from Naval Air Rework Facility Production and Planning Notices dated 17 June 1971 and 26 July 1971 respectively. The forecasted requirements were converted into two separate production vectors (Y's).

Penalty costs were computed for each type of engine and aircraft reworked at NARF North Island using information supplied by MSDO. Aircraft penalty costs were computed by prorating the total cost of the aircraft over its estimated service life. The resulting penalty cost was in units of dollars per day in shop. A similar computation was desired for engine penalty costs. However, an estimated service life for each engine was not available. When engines are reworked by the NARF, engine components are replaced as necessary resulting in a "new" engine. For this reason the service life of all engines was assumed to be ten years. Table V shows the computed penalty costs. For convenience Tables V through XI are located at the end of this chapter.

A constant was also needed for converting direct man hours into a dollar cost. As previously defined in Chapter III, this is the value of p in the vector of prices, P.

A computer program was developed to compute the average labor cost. The computer program is shown in Appendix J. The results were as follows:

	$\underline{p}$ (in dollars per man-hour)
engines	6.04
aircraft	6.17

B. COMPUTATIONAL PROCEDURES

To solve the engine and aircraft linear programs Mathematical Programming System/360 (MPS/360) was used. This IBM applications program is composed of a very complex set of computational procedures. A subset of these procedures are used to solve linear programs. However, the user must construct a control program that orders a specific sequence of procedures to be executed by the computer. A detailed knowledge of Refs. 6 and 7 is necessary to use MPS/360.

In addition to preparing a control program for MPS/360, the linear program input data must be in a special format described in Chapter 4 of Ref. 7. To assist in preparing data in this format a computer program was prepared. This program takes specified input data and produces a deck of punched cards for MPS/360 in the required format. This program is documented in Appendix K.

C. ENGINE MODEL

1. Input Data

The sample data used for the engine model is listed in Table VI. Each row of the T_1 submatrix of matrix T is associated with a specific type of engine and type of work done on that engine. The production vector Y is associated with each row of the T_1 matrix because Y specifies the required number of man-hours of NORM for the particular type of work. The linear program must then choose the process (columns of matrix T) by which the cost of the work will be a minimum. If only one process is available and production is required then the linear program must select that process.

The total amount of resources available are shown as rows R1, R2 and R3. These values form the resource vector R.

The model was solved for three different, but closely related, objective functions. The values listed in columns OBJ1, OBJ2 and OBJ3 of Table VI are calculated as follows:

$\text{OBJ1} = P^T \cdot T_2 + C^T \cdot T_3$ = total cost per unit output of NORM (total cost = resource cost + penalty cost)

$\text{OBJ2} = P^T \cdot T_2$ = resource cost per unit output of NORM

$\text{OBJ3} = C^T \cdot T_3$ = penalty cost per unit output of NORM

In Table VI OBJ1 = OBJ2 + OBJ3 except in cases where numbers were rounded off by the computer.

Recall from Chapter III that the T_1 matrix consisted of a 1 in only one row of each column of T_1 and all the other elements of that column were zeros. The row in which the 1 belongs is given in Table VI in the T1 ROW column. The columns in Table VI labeled R1, R2 and R3 are the three elements of the T_2 matrix described in Chapter III. These three values correspond to the amount of each resource required <u>per unit output</u> of NORM for that particular process. The column labeled T3 in Table VI is the number of days in shop <u>per unit output</u> of NORM for that particular process. The values for vector C are found in Table V. However, the penalty costs <u>per unit output</u> of NORM may be found directly in the OBJ3 column of Table VI. Recall that $OBJ = C^T \cdot T_3$.

2. <u>Solutions</u>

The solutions to the three linear programs (each with a different objective function) using the engine model data are presented in Tables VII, VIII and IX. The optimal solution for each problem is shown in the first column of each Table. The actual choice of processes selected by MPS/360 is not shown, because as stated in Chapter III, these "processes" do not represent actual physical procedures which are used by the NARF. The processes do give an estimate of the total production cost for a specified production vector. The anticipated change in production costs will be the foundation for the cost justification of WIPICS.

It is necessary to explain the relationship between the optimal solutions presented in Tables VII and VIII. The solutions are nearly identical. This result is not unexpected as can be seen in Table VI. The magnitude of OBJ2 (resource cost) is much larger than OBJ 3 (penalty cost) in every case. By attempting to minimize the total cost of production (resource cost + penalty cost) the linear program arrived at a solution that also happened to minimize resource cost. This is not an uncommon result when one component of an objective function is much larger than another component. One might also observe that the value of the objective function in Table VII is approximately 90% resource cost and 10% penalty cost. This is another reason for the two solutions to be similar.

In Table IX the solution to minimization of penalty cost turns out to be very costly in terms of required resource costs. An increase of nearly $300,000 in resource cost for a penalty cost reduction of only $20,000 is the difference between Tables VII and IX. The obvious conclusion is that unless penalty cost is higher than presently calculated, the resource cost will nearly always determine the optimal solution. A discussion of alternate methods of determining penalty cost will be discussed in Chapter V.

3. Sensitivity Analysis

Using procedures described in the MPS/360 references a parametric study was done on the resources DMHR, DML$ and DOH$. Since the optimal solution was obtained with

unconstrained resources available it is not necessary to increase any resource. The optimal solution shown in Tables VII, VIII and IX will remain the same for all values of resources greater than those shown in Table VI. It is possible to obtain different solutions to each model by decreasing the resources one at a time until the linear program becomes infeasible. Infeasible means that there is not enough of some resource in R to accomplish the work specified by the production vector Y. In other words, a conflict exists between some of the constraints in the model and no solution exists for that linear program. The results of decreasing each resource (one at a time) are shown in the parametric studies section of each solution table.

If the available quantity of a resource is reduced in value until the entire quantity is consumed in production, then any further reduction below that value will result in an increased cost in the objective function. In linear programming terms the shadow price of the dual variable associated with that resource constraint is frequently positive when the constraint is binding. The increased cost of production as described above can be observed in Tables VII through IX.

An interesting result appears in Tables VII and VIII. When the DMHR resource was reduced, the DOH$ resource was also reduced in the new solution (but the DML$ resource use increased). The explanation for this effect is found in Table III. The correlation coefficient between DMHR and

DOH$ is +.99032. This high correlation suggests that overhead costs at the NARF are applied to each job based on the DMHR for the job. This is the way overhead costs are applied at NARF, North Island. Therefore, a decrease in available DMHR could also cause a reduction in DOH$ used.

In summary, the parametric studies on the resource vector indicate the minimum amount of available resources required for a feasible linear program. However, if all three resources are reduced to their respective minimum values at the <u>same time</u> an infeasible linear program will result. The reason for this may be seen in Table VII. In reducing DMHR the requirement for DML$ is <u>increased</u>. The parametric study results as shown were used to constrain only a single resource and the two remaining resources were considered available in unlimited amounts. Other types of parametric studies are available if desired. For example, all three resources may be changed in fixed proportions. The MPS/360 Users Manual (Ref. 7) discusses available post-optimal procedures.

D. AIRCRAFT MODEL

1. <u>Input Data</u>

The sample data for the aircraft model is very similar to the engine data described previously. Each row of T_1 is associated with a particular type of aircraft and type of work. The procuction vector Y and the resource vector R are the same as described in the engine model. The data is listed in Table X.

The aircraft model consisted of a single objective function which was calculated as follows:

$$OBJ = P^T T_2 + C^T T_3 = \text{total cost per unit output of NORM}$$

(total cost = resource cost + penalty cost)

The remaining columns in Table X are identical in meaning to the engine model descriptions for Table VI.

2. Solution

The solution of the aircraft model is presented in Table XI. Due to the manner in which the data was aggregated the aircraft model consisted of only one process for each type of job in the production vector Y. The T_1 matrix consisted of 70 rows and 70 columns. Linear programming is not necessary to solve a problem of this type. There are no choices to make among processes, therefore, the activity level vector Z must be identical to the production vector Y (because T_1 is an identity matrix). The solution in Table XI does give an estimated total production cost associated with the production vector Y which can be used for budget estimates. A discussion concerning the aggregation problems for the aircraft data is presented in Chapter V. No sensitivity analysis was conducted on the aircraft model because the amounts of resources required at the optimal (unconstrained resources) solution also represent the minimum resources required for a feasible solution to the linear program. This result occurs because there is only one process for each job type in the production vector Y.

Table V

Penalty Costs

Aircraft Program

Type Aircraft	Unit Cost	Service Life (months)	Penalty Costs ($/day in shop)
C-2A	$10,742,000	105	280.29
E-2B	10,742,000	105	280.29
F-4B	2,756,000	80	94.38
RF-4B	2,699,000	96	77.02
F-4J	2,492,000	80	85.34
RF-8G	1,300,000	75	47.48
F-8H	1,352,000	54	68.59
F-8J	1,302,000	54	66.05
RH-3A	1,508,000	80	51.64
SH-3A	1,278,000	80	43.76
SH-3D	1,064,000	80	36.43
CH-46A	1,083,000	80	37.08
UH-46A	834,000	80	28.56
CH-46D	1,083,000	80	37.08
UH-46D	834,000	80	28.56
CH-46F	1,083,000	80	37.08
CH-53A	1,453,000	96	41.46
CH-53D	1,453,000	96	41.46

Table V - Penalty Costs (continued)

Engine Program *

Type Engine	Unit Cost	Penalty Costs ($/day in shop)
J79-GE-8B	$225,000	61.64
J79-GE-8C	225,000	61.64
J79-GE-10	180,000	49.32
T58-GE-8B	65,000	17.81
T58-GE-8F	65,000	17.81
T58-GE-10	65,000	17.81
T64-GE-6B	125,000	34.25
T64-GE-413	167,000	45.25
J57 (all models)	210,000	57.53
T56 (all models)	98,600	27.01

* a 10 year service life was assumed for all engines.

ROW NO.	TYPE/WORK CODE	PRODUCTION VECTOR (Y)
1	51/01	435.00
2	51/04	0.0
3	53/01	0.0
4	53/01	6675.00
5	53/04	0.0
6	55/01	3290.00
7	55/04	0.0
8	56/04	0.0
9	56/03	160.00
10	57/01	800.00
11	63/01	3500.00
12	63/04	0.0
13	65/01	3500.00
14	65/01	3600.00
15	65/01	1425.00
16	65/03	0.0
17	65/03	5075.00
18	65/03	0.0
19	65/04	3450.00
20	65/04	3920.00
21	65/04	0.0
22	65/04	5250.00
23	66/04	0.0
24	67/01	400.00
25	67/04	990.00
26	69/01	6580.00
27	69/03	2025.00
28	69/04	990.00
29	72/04	0.0
30	74/03	600.00
31	74/04	5100.00
32	73/03	0.0
33	75/03	1305.00
34	75/03	440.00
35	73/04	0.0
36	73/04	0.0
37	75/04	1860.00
38	75/04	2760.00
39	81/01	4320.00
40	81/03	600.00
41	81/04	5985.00
42	82/03	8625.00
43	83/02	50700.00
44	83/02	0.0
45	83/02	31250.00
46	83/04	6930.00
47	84/01	16100.00

ROW NO.	RESOURCE	RESOURCE VECTOR (R)
R1	DMHR	200000.00
R2	DML$	2500000.00
R3	DOH$	1600000.00

Table VI. Engine Model Input Data

COLUMN NO.	OBJ1	OBJ2	OBJ3	T1 ROW	R1	R2	R3	T3
1	25.325	23.955	1.372	1	0.910	11.530	6.926	0.077
2	16.187	14.543	1.644	2	0.921	2.304	6.675	0.092
3	20.456	17.696	2.761	3	0.840	5.875	6.747	0.155
4	24.699	23.285	1.415	4	0.939	10.481	7.131	0.079
5	21.354	19.446	1.908	5	0.856	8.135	6.141	0.107
6	28.626	26.807	1.819	6	1.196	10.532	9.053	0.102
7	16.395	14.815	1.580	7	0.762	4.535	5.681	0.089
8	41.069	27.641	13.428	8	0.603	20.474	3.526	0.497
9	58.153	45.323	12.830	9	1.200	31.750	6.325	0.475
10	44.125	39.061	5.064	10	1.212	21.719	10.019	0.188
11	35.674	33.045	2.629	11	1.120	17.369	8.912	0.077
12	20.841	18.256	2.585	12	1.146	3.024	8.313	0.075
13	31.741	29.432	2.309	13	0.996	15.403	8.014	0.067
14	40.934	39.124	1.810	13	0.985	25.858	7.316	0.053
15	36.803	35.018	1.786	13	1.066	20.825	7.756	0.052
16	34.803	30.368	4.435	14	1.047	15.643	8.400	0.129
17	37.496	33.844	3.652	14	1.288	16.466	9.599	0.107
18	38.270	35.078	3.192	14	1.227	18.595	9.076	0.093
19	38.656	34.870	3.787	15	1.158	19.307	8.571	0.111
20	33.024	28.681	4.343	15	0.906	15.983	7.229	0.127
21	38.310	34.500	3.810	15	1.036	20.564	7.676	0.111
22	27.758	24.761	2.997	15	1.164	9.038	8.690	0.087
23	36.983	32.583	4.400	16	1.233	16.070	9.067	0.128
24	46.711	44.425	2.286	17	0.850	32.658	6.632	0.067
25	32.310	29.431	2.879	18	0.663	20.372	5.056	0.084
26	58.557	55.467	3.090	19	0.869	43.607	6.613	0.068
27	32.755	28.263	4.493	20	0.816	17.239	6.095	0.098
28	81.150	74.698	6.452	21	1.061	60.165	8.125	0.141
29	57.252	52.705	4.548	22	1.194	35.754	9.737	0.079
30	47.263	38.838	8.425	23	1.074	23.969	8.381	0.146
31	37.245	32.535	4.710	24	1.159	16.483	9.054	0.082
32	26.648	22.004	4.644	25	1.133	5.959	9.199	0.081
33	26.586	22.046	4.540	25	0.727	12.235	5.422	0.079
34	25.173	21.398	3.775	25	0.693	12.297	4.917	0.066
35	29.052	25.482	3.570	26	1.141	9.722	8.866	0.062
36	23.809	19.768	4.042	26	0.925	7.396	6.786	0.070
37	28.829	24.213	4.616	26	0.903	11.855	6.905	0.080
38	31.330	27.211	4.119	26	0.938	15.163	6.381	0.072
39	31.292	29.626	1.666	27	1.042	14.883	8.451	0.094
40	20.931	19.308	1.623	28	0.799	8.082	6.399	0.091

Table VI. Engine Model Input Data (Continued)

COLUMN NO.	OBJ1	OBJ2	OBJ3	T1 ROW	R1	R2	R3	T3
41	17.245	15.761	1.484	29	0.583	8.155	4.088	0.083
42	30.158	27.723	2.435	30	1.600	5.042	13.017	0.137
43	32.001	30.157	1.844	31	1.114	14.898	8.528	0.104
44	26.818	25.256	1.562	31	1.088	10.119	8.566	0.088
45	32.332	30.989	1.343	31	0.954	18.171	7.058	0.075
46	24.703	22.319	2.384	32	1.082	7.692	8.095	0.134
47	28.841	27.337	1.503	33	0.975	13.958	7.490	0.084
48	20.255	18.069	2.186	34	0.938	5.341	7.064	0.123
49	25.665	24.163	1.502	35	0.983	10.587	7.641	0.084
50	28.008	26.658	1.350	35	0.913	13.998	7.143	0.076
51	31.497	30.379	1.118	36	0.910	18.230	6.655	0.063
52	20.595	18.650	1.946	37	0.960	5.382	7.472	0.109
53	22.871	20.966	1.905	37	0.903	8.904	6.609	0.107
54	22.814	21.168	1.646	37	0.912	8.690	6.969	0.092
55	28.178	26.479	1.699	37	0.926	14.631	6.252	0.095
56	28.243	26.859	1.384	38	0.931	14.072	7.163	0.078
57	31.030	29.719	1.311	38	0.965	16.589	7.304	0.074
58	34.293	33.046	1.248	38	0.884	21.465	6.243	0.070
59	26.276	23.935	2.341	39	1.093	8.505	8.829	0.131
60	26.547	24.586	1.962	39	1.121	9.811	8.004	0.110
61	29.557	27.604	1.954	39	1.127	12.774	8.021	0.110
62	28.085	25.440	2.645	40	1.150	9.785	8.710	0.149
63	27.071	25.148	1.923	40	1.080	10.531	8.095	0.108
64	28.756	22.542	6.213	41	1.122	7.317	8.450	0.101
65	25.530	21.590	3.940	42	1.067	6.977	8.170	0.064
66	24.725	21.303	3.423	42	0.926	9.203	6.507	0.056
67	26.064	23.614	2.450	43	1.008	9.417	8.105	0.040
68	26.124	24.175	1.949	43	1.010	10.213	7.859	0.032
69	26.880	25.068	1.812	43	0.978	11.529	7.633	0.029
70	25.036	23.113	1.923	43	0.978	10.449	6.758	0.031
71	14.044	11.069	2.976	44	0.701	1.259	5.577	0.048
72	23.924	22.084	1.840	45	0.965	8.771	7.483	0.037
73	24.101	22.292	1.809	45	0.922	9.156	7.564	0.037
74	23.133	21.472	1.660	45	1.011	7.807	7.558	0.034
75	25.925	24.288	1.637	45	1.027	10.134	7.951	0.033
76	24.489	22.928	1.561	45	0.972	10.165	6.893	0.032
77	23.955	19.908	4.047	46	1.014	5.968	7.814	0.082
78	23.938	19.725	4.213	46	0.987	6.506	7.259	0.085
79	26.011	21.334	4.677	47	1.065	6.763	8.136	0.095
80	27.863	24.126	3.736	47	1.288	6.349	10.001	0.076
81	23.158	20.124	3.033	47	0.980	7.066	7.140	0.062

Table VI. Engine Model Input Data (Continued)

Objective Function = $P^T T_2 Z + C^T T_3 Z$

	Optional Solution	Parametric Studies on Optional Solution		
	Unconstrained Resources	Minimizing DMHR Used	Minimizing DML$ Used	Minimizing DOH$ Used
Objective Function Value	5,227,836	5,379,854	5,364,833	5,419,475
Resource Cost	4,742,057	4,887,049	4,836,428	4,939,728
Penalty Cost	485,779	492,805	528,405	479,747
DMHR Used	187,917	183,674 *	196,311	185,617
DML$ Used	2,215,889	2,401,905	2,122,367 *	2,471,487
DOH$ Used	1,391,150	1,375,753	1,528,343	1,347,115 *

* represent minimum amount of resource required for a feasible solution (assuming other two resources are unlimited)

Table VII

Engine Model Solution for Total Cost Objective Function

Objective Function = $P^T \cdot T_2 \cdot Z$

	Optimal Solution	Parametric Studies on Optional Solution		
	Unconstrained Resources	Minimizing DMHR Used	Minimizing DML$ Used	Minimizing DOH$ Used
Objective Function Value	4,742,043	4,887,036	4,836,423	4,939,717
Resource Cost	4,742,043	4,887,036	4,836,423	4,939,717
Penalty Cost	485,779	492,805	528,405	479,747
DMHR Used	187,917	183,674 *	196,311	185,617
DML$ Used	2,215,889	2,401,905	2,122,367 *	2,471,487
DOH$ Used	1,391,150	1,375,753	1,528,343	1,347,115 *

* represent minimum amount of resource required for feasible solution (assuming other two resources are unlimited)

Table VIII

Engine Model Solution for Resource Cost Objective Function

Objective Function = $C^T \cdot T_3 \cdot Z$

	Optimal Solution	Parametric Studies on Optimal Solution		
	Unconstrained Resources	Minimizing DMHR Used	Minimizing DML$ Used	Minimizing DOH$ Used
Objective Function Value	462,715	487,152	528,410	479,764
Resource Cost	5,028,894	4,986,177	4,836,428	4,939,728
Penalty Cost	462,715	487,152	528,410	479,764
DMHR Used	188,439	183,674 *	196,311	185,617
DML$ Used	2,471,565	2,456,646	2,122,367 *	2,471,487
DOH$ Used	1,419,158	1,420,141	1,528,343	1,347,115 *

* represent minimum amount of resource required for a feasible solution (assuming other two resources are unlimited)

Table IX

Engine Model Solution for Penalty Cost Objective Function

ROW NO.	TYPE/WORK CODE	PRODUCTION VECTOR (Y)
1	10/41	0.0
2	10/42	9300.00
3	11/32	0.0
4	11/33	12000.00
5	11/34	81200.00
6	21/11	0.0
7	21/21	93600.00
8	21/22	32800.00
9	21/41	0.0
10	21/62	0.0
11	22/13	23500.00
12	22/14	0.0
13	22/22	0.0
14	22/23	0.0
15	22/24	0.0
16	22/25	0.0
17	22/42	73200.00
18	22/43	89000.00
19	22/44	13800.00
20	22/45	70000.00
21	22/52	0.0
22	22/53	0.0
23	22/54	0.0
24	22/55	0.0
25	22/63	0.0
26	22/64	0.0
27	22/65	0.0
28	22/66	0.0
29	23/34	0.0
30	25/11	0.0
31	25/21	14000.00
32	25/22	0.0
33	25/61	0.0
34	26/11	0.0
35	26/12	0.0

Table X. Aircraft Model Input Data

ROW NO.	TYPE/WORK CODE	PRODUCTION VECTOR (Y)
36	26/21	0.0
37	26/61	0.0
38	26/62	0.0
39	27/12	0.0
40	27/22	0.0
41	27/23	0.0
42	27/61	0.0
43	27/63	0.0
44	31/20	0.0
45	32/22	4500.00
46	32/23	4200.00
47	32/24	0.0
48	33/21	0.0
49	33/23	0.0
50	34/33	0.0
51	34/34	34000.00
52	34/35	46800.00
53	34/36	0.0
54	35/12	0.0
55	35/22	0.0
56	41/22	4500.00
57	41/23	4400.00
58	41/43	0.0
59	42/22	32000.00
60	42/42	0.0
61	43/21	18000.00
62	44/23	0.0
63	45/22	0.0
64	48/21	14700.00
65	48/22	29400.00
66	48/71	0.0
67	48/72	0.0
68	48/82	0.0
69	49/21	44000.00
70	49/71	0.0

ROW NO.	RESOURCE	RESOURCE VECTOR (R)
R1	DMHR	800000.00
R2	DML$	2000000.00
R3	DOH$	5000000.00

Table X. Aircraft Model Input Data (Continued)

COLUMN NO.	OBJ	T1 ROW	R1	R2	R3	T3
1	17.941	1	1.030	2.073	6.475	0.011
2	17.919	2	0.992	2.847	6.046	0.010
3	18.973	3	1.030	2.803	6.979	0.010
4	19.425	4	1.088	2.832	7.283	0.009
5	18.642	5	1.119	2.543	6.652	0.009
6	18.871	6	1.028	4.978	6.572	0.011
7	17.125	7	1.065	2.813	7.000	0.009
8	16.663	8	1.079	2.576	6.791	0.007
9	14.961	9	1.023	1.435	6.429	0.009
10	15.735	10	1.038	2.164	6.486	0.008
11	15.513	11	0.997	2.400	5.931	0.011
12	16.677	12	1.014	2.973	6.352	0.012
13	15.487	13	1.059	1.950	6.376	0.007
14	15.276	14	0.979	2.361	6.087	0.008
15	15.826	15	1.014	2.467	6.254	0.009
16	16.093	16	1.026	2.353	6.485	0.010
17	15.661	17	0.997	1.649	7.148	0.008
18	15.629	18	1.029	1.952	6.553	0.008
19	15.944	19	1.052	2.038	6.562	0.009
20	16.721	20	1.108	1.983	6.963	0.010
21	16.217	21	1.020	2.443	6.577	0.010
22	15.619	22	1.026	2.021	6.490	0.008
23	15.055	23	1.034	1.610	6.304	0.008
24	16.758	24	1.108	2.275	6.770	0.009
25	13.223	25	0.865	2.008	5.148	0.008
26	15.649	26	1.022	2.328	6.262	0.008
27	18.000	27	1.188	2.374	7.239	0.011
28	16.033	28	1.005	2.812	6.321	0.007
29	11.958	29	0.783	1.240	5.299	0.006
30	15.791	30	0.959	2.403	6.514	0.015
31	16.053	31	1.018	2.571	6.495	0.011
32	14.020	32	0.868	2.867	5.144	0.010
33	16.943	33	1.091	2.907	6.534	0.012
34	16.553	34	1.028	2.461	6.677	0.016
35	15.980	35	0.979	2.161	6.787	0.014

Table X. Aircraft Model Input Data (Continued)

COLUMN NO.	OBJ	T1 ROW	R1	R2	R3	T3
36	15.101	36	0.943	2.297	6.315	0.010
37	16.364	37	0.993	2.542	6.998	0.010
38	19.732	38	1.222	3.932	7.482	0.011
39	15.867	39	1.026	2.292	6.378	0.018
40	13.234	40	0.842	1.724	5.901	0.009
41	17.017	41	1.215	1.508	7.534	0.010
42	16.355	42	1.071	2.788	6.504	0.010
43	17.678	43	1.204	2.498	7.207	0.011
44	15.694	44	0.955	2.824	6.409	0.011
45	21.291	45	1.153	4.847	8.573	0.015
46	17.919	46	1.140	2.948	7.212	0.014
47	14.515	47	0.761	3.669	5.590	0.011
48	12.693	48	0.771	2.539	4.798	0.014
49	13.960	49	0.868	1.700	6.377	0.012
50	20.948	50	1.395	2.478	9.050	0.019
51	15.541	51	1.020	2.193	6.509	0.012
52	13.731	52	0.893	2.111	5.595	0.012
53	16.193	53	1.057	2.332	6.735	0.014
54	16.633	54	0.948	3.481	6.616	0.019
55	15.174	55	0.962	2.318	6.449	0.013
56	17.048	56	1.021	3.870	6.394	0.013
57	16.030	57	0.980	3.392	6.084	0.014
58	14.755	58	0.924	1.846	6.717	0.013
59	16.370	59	1.013	3.247	6.363	0.014
60	16.303	60	1.041	2.336	7.042	0.014
61	14.463	61	0.857	3.053	5.626	0.014
62	16.065	62	0.954	3.356	6.483	0.012
63	16.065	63	0.888	4.054	6.111	0.015
64	20.321	64	1.072	5.616	7.553	0.013
65	19.364	65	1.082	4.557	7.549	0.014
66	16.846	66	0.889	4.672	6.345	0.008
67	18.359	67	0.957	5.921	6.153	0.009
68	17.534	68	1.035	4.545	6.123	0.012
69	15.430	69	0.963	3.131	5.956	0.010
70	16.447	70	1.018	3.136	6.662	0.009

Table X. Aircraft Model Input Data (Continued)

Table XI

Aircraft Model Solution

for Total Cost Objective Function

Optimal Solution with Unconstrained Resources

Objective Function Value	12,400,594
Resource Cost	11,691,151
Penalty Cost	709,443
DMHR Used	775,635
DML$ Used	1,921,528
DOH$ Used	4,983,956

V. CONCLUSIONS AND AREAS FOR FUTURE STUDY

A. CONCLUSIONS

The sample problem solutions in Chapter IV answered some basic questions concerning the operation of a NARF. In the beginning of this study, it was not known if tradeoffs existed between resources used by the NARF. The solutions in Table VII showed that the ranges of the resources used were rather limited for feasible solutions. For example, in Table VII the reduction in use of DMHR to a minimum resulted in a 2.3% decrease from the optimal unconstrained solution. Similarly minimizing DML$ and DOH$ produced resource reductions of 4.2% and 11.2% respectively. Associated with minimizing DMHR, DML$ and DOH$ (one at a time) the total cost of production increased approximately 2 to 3% in each case. This demonstrated that the available tradeoffs among resources are small in the linear economic model. This result was not unexpected because of the similarity among processes estimated for each type of work.

The total cost objective function is composed of two components, resource cost and penalty cost. In the sample problems considered the total cost was approximately 90% resource cost and 10% penalty cost. This fact caused the total cost optimal solution to be the same as the resource cost optimal solution. Alternate methods of determining penalty costs are discussed in Section B of this chapter.

The sample problems solved in Chapter IV predicted budget requirements for a specific calender quarter, based on forecasted production requirements. To draw a meaningful conclusion about the validity of the model these results must be compared with the actual production work in that quarter and the actual quantities of resources consumed. This type of comparison should be made over a series of production periods of varying length to determine the accuracy of the model. This validation was not completed in time for inclusion in this thesis. Since this model is only a part of a continuing project at the Naval Postgraduate School, additional results are expected in the near future.

B. AREAS FOR FUTURE STUDY

1. Expanding Number of Observations

As mentioned in Chapter II and IV, special problems were encountered with the aircraft model sample data because of the small number of raw data observations. Based on discussions with NARF and MSDO personnel a decision was made to consider work done in different "cycles" (a cycle is the number of times the aircraft has been through the NARF for the same type of work) as different types of work. The resulting aircraft model had 70 separate processes with only 365 observations to estimate the processes. In comparison the engine model had 81 separate processes and 1532 observations. Forty-one out of 70 processes in the aircraft model had three or fewer observations. The engine

program had only 11 out of 81 processes with three or fewer observations. The magnitude of this problem could be reduced by increasing the number of observations in each model. However, an upper limit should be placed on the age of an observation to ensure the processes that are estimated reflect the current technology at the NARF.

2. Penalty Costs

As seen in Chapter IV the penalty costs had very little effect in determining the solution to the engine model. The reason was that the resource cost was at least five times as large as the penalty cost when the objective function was formulated. (See Table VI for examples). It could easily be argued that the present method used to determine the penalty costs does not approximate the opportunity cost (for the Navy) of keeping an engine (or aircraft in the shop for an extra day.

One alternate method by which the penalty cost might be computed is:

$$\text{penalty cost} = \frac{\begin{array}{c}\text{procurement cost}\\ + \\ \text{total maintenance cost over lifetime}\end{array}}{\text{lifetime}}$$

Certainly there is not a single procedure to determine penalty cost that cannot be criticized in some area. However, this problem must be addressed and resolved if an accurate cost-effectiveness analysis of WIPICS is to result.

3. Production Vector

In the linear economic model formulated in Chapter III the production vector (Y) is assumed as a given input to the model. In reality, the production demanded of the NARF each calender quarter is the result of a complex negotiation procedure between NAVAIRSYSCOM representatives and NARF representatives. Each component (Y_i) of the production vector may be considered a random variable whose value depends on the negotiation results. It is possible to construct a mathematical model of the negotiation procedures that would generate (predict) production vectors for use in the linear economic model. Further consideration should be given to a model of this type in lieu of using fixed production vectors based on historical data.

4. Management Constraints

One area in which extensions of the linear economic model would be beneficial is the incorporation of constraints for management practices. For example, there is a fixed manpower ceiling at the NARF for the permanent (employed at the NARF for more than one year) labor force. Suppose a given production vector (Y) utilizes the entire DMHR resource making the model infeasible. A temporary labor force could be hired from the surrounding community and be used to make up the deficit DMHR. The model becomes feasible.

Another possible extension exists in the pricing of labor. The management policy at the NARF could dictate a maximum allowable percentage of overtime per quarter. The overtime man-hours could be modeled as an additional constraint and priced according to current union contract agreements.

Further study is required to determine the effects of management decisions (or management constraints such as union contracts) on the linear economic model. Modifications and/or additional constraints may be necessary.

APPENDIX A

Correlation Program for Raw Data

A. Purpose: The purpose of this program is to calculate and print a table of correlation coefficients between NORM, DMHR, DLB$, DML$, DOH$, NIFR and NDAY. A table of means and standard deviations is printed for each of the above variables.

B. Inputs: The first card is a control card which is then followed by the raw data cards. All fields are right adjusted unless otherwise specified. The abbreviation "CC" refers to card columns and will be used in all Appendices. The format of the cards follows:

Control Card:	CC 1-4	number of raw data cards
	CC 5-8	0001 for aircraft
		0002 for engines

Raw data card:	CC 1-2	engine/aircraft code
	CC 3	blank
	CC 4-10	Identification number
	CC 11	blank
	CC 12-13	work code
	CC 14	blank
	CC 15-18	induction date
	CC 19	blank
	CC 20-23	production date

CC 24	blank
CC 25-29	NORM
CC 30	blank
CC 31-35	airframe change man-hours (blank for engine cards)
CC 36	blank
CC 37-41	DMHR
CC 42	blank
CC 43-48	DLB$
CC 49	blank
CC 50-55	DML$
CC 56	blank
CC 57-62	DOH$
CC 63	blank
CC 64-68	NIFR

The NIFR is expressed as an integer but the last two digits are cents. For example a NIFR of 1482 means $14.82. All other dollar values are in whole dollars.

C. Outputs: A table of correlation coefficients and a table of means and standard deviations for each variable are printed.

D. Special Notes:

(1) Subroutines CORRE, LOC and MSTR are from the IBM Scientific Subroutine Package. (Input to the CORRE subroutine is a linear array vice a matrix to simplify dimension requirements).

(2) The maximum number of data cards that the program can handle is 1600. Capacity may be increased by changing the DIMENSION and DATA statements.

```
C*******************************************************************************
C
C
C        CORRELATION PROGRAM FOR RAW DATA
C
C             INPUT REQUIRED:
C                 (1) NUMBER OF DATA CARDS
C                     KN=1 FOR AIRCRAFT
C                     KN=2 FOR ENGINES
C                     FORMAT (2I4)
C                 (2) RAW DATA CARDS
C
C
C*******************************************************************************
      DIMENSION R(7,7),RA(7,7),RX(7,7)
      DIMENSION XBAR(7),STD(7),B(7),D(7),T(7),X(11200),A(7),AT(2)
      REAL*8 AT
      DATA X,R,RX,RA,XBAR,STD,B,D,T/11382*0.0/
      DATA MSA,MSR,IO,M/1,0,1,7/
      DATA AT(1),AT(2)/'AIRCRAFT',' ENGINES'/
      DATA A/'NORM','DMHR','DLB$','DML$','DOH$','NIFR','NDAY'/
      READ (5,9000) N,KN
 9000 FORMAT (2I4)
      DO 1000 I=1,N,1
      READ (5,9001) ID,JD,  X(I),X(I+N),X(I+2*N),X(I+3*N),
     6X(I+4*N),X(I+5*N)
 9001 FORMAT (13X,2I5,F6.0,6X,F6.0,3F7.0,F6.0)
      NDAYS=JD-ID
      IF (NDAYS.GE.635) NDAYS=NDAYS-635
      X(I+6*N)=FLOAT(NDAYS)
 1000 CONTINUE
      CALL CORRE (N,M,IO,X,XBAR,STD,RX,R,B,D,T)
      CALL MSTR (R,RA,M,MSA,MSR)
      WRITE (6,9500) AT(KN), A
 9500 FORMAT ('1',////,45X,'CORRELATION FOR RAW DATA - ',A8,
     6////,29X,7A10,/)
      DO 2000 I=1,7,1
      WRITE (6,9501) A(I),(RA(I,J),J=1,7)
 9501 FORMAT (' ',25X,A4,7F10.5,/)
 2000 CONTINUE
      WRITE (6,9502)
 9502 FORMAT (' ',////,35X,'MEAN', 4X,'STD DEV',/)
      DO 3000 I=1,7,1
      WRITE (6,9503) A(I),XBAR(I),STD(I)
 9503 FORMAT (' ',25X,A4,2F10.2,/)
 3000 CONTINUE
      STOP
      END
```

```
      SUBROUTINE CORRE (N,M,IO,X,XBAR,STD,RX,R,B,D,T)
      DIMENSION X(1),XBAR(1),STD(1),RX(1),R(1),B(1),D(1),T(1)
      DO 100 J=1,M
      B(J)=0.0
  100 T(J)=0.0
      K=(M*M+M)/2
      DO 102 I=1,K
  102 R(I)=0.0
      FN=N
      L=0
  105 DO 108 J=1,M
      DO 107 I=1,N
      L=L+1
  107 T(J)=T(J)+X(L)
      XBAR(J)=T(J)
  108 T(J)=T(J)/FN
      DO 115 I=1,N
      JK=0
      L=I-N
      DO 110 J=1,M
      L=L+N
      D(J)=X(L)-T(J)
  110 B(J)=B(J)+D(J)
      DO 115 J=1,M
      DO 115 K=1,J
      JK=JK+1
  115 R(JK)=R(JK)+D(J)*D(K)
  205 JK=0
      DO 210 J=1,M
      XBAR(J)=XBAR(J)/FN
      DO 210 K=1,J
      JK=JK+1
  210 R(JK)=R(JK)-B(J)*B(K)/FN
      JK=0
      DO 220 J=1,M
      JK=JK+J
  220 STD(J)= SQRT( ABS(R(JK)))
      DO 230 J=1,M
      DO 230 K=J,M
      JK=J+(K*K-K)/2
      L=M*(J-1)+K
      RX(L)=R(JK)
      L=M*(K-1)+J
      RX(L)=R(JK)
      IF(STD(J)*STD(K)) 225, 222, 225
  222 R(JK)=0.0
```

```
      GO TO 230
  225 R(JK)=R(JK)/(STD(J)*STD(K))
  230 CONTINUE
      FN=SQRT(FN-1.0)
      DO 240 J=1,M
  240 STD(J)=STD(J)/FN
      L=-M
      DO 250 I=1,M
      L=L+M+1
  250 B(I)=RX(L)
      RETURN
      END

      SUBROUTINE LOC(I,J,IR,N,M,MS)
      IX=I
      JX=J
      IF(MS-1) 10,20,30
   10 IRX=N*(JX-1)+IX
      GO TO 36
   20 IF(IX-JX) 22,24,24
   22 IRX=IX+(JX*JX-JX)/2
      GO TO 36
   24 IRX=JX+(IX*IX-IX)/2
      GO TO 36
   30 IRX=0
      IF(IX-JX) 36,32,36
   32 IRX=IX
   36 IR=IRX
      RETURN
      END

      SUBROUTINE MSTR(A,R,N,MSA,MSR)
      DIMENSION A(1),R(1)
      DO 20 I=1,N
      DO 20 J=1,N
      IF(MSR) 5,10,5
    5 IF(I-J) 10,10,20
   10 CALL LOC(I,J,IR,N,N,MSR)
      IF(IR) 20,20,15
   15 R(IR)=0.0
      CALL LOC(I,J,IA,N,N,MSA)
      IF(IA) 20,20,18
   18 R(IR)=A(IA)
   20 CONTINUE
      RETURN
      END
```

APPENDIX B

Cluster Data Program

A. Purpose: The purpose of this program is to calculate and provide a punched card deck for input to the distance matrix program in Appendix C. Each input data card is normalized to a unit vector for the five variables NORM, DMHR, DML$, DOH$ and NDAY. Then each variable is standardized by the following transformation:

$$\frac{x_{ij} - \mu_j}{\sigma_j}$$

where x_{ij} = $i^{\underline{th}}$ observation on the $j^{\underline{th}}$ variable

i = 1,...,n

j = 1,...,5

n = number of data cards

σ_j = standard deviation of the $j^{\underline{th}}$ variable

μ_j = mean of the $j^{\underline{th}}$ variable

B. Inputs: The first card is a control card which is followed by a deck of aggregated data cards. The aggregated data cards are described in Appendix F.

Control Card: CC 1-3 number of aggregated data cards

Aggregated data card: the format is specified in Appendix F.

C. Outputs: Listings of input data, normalized data and normalized/standardized data and a deck of punched cards

for input to the distance matrix program in Appendix C are provided. The punched output deck is in the following format:

CC 1-2	engine/aircraft type code
CC 3	blank
CC 4-10	Identification code
CC 11	blank
CC 12-13	work code
CC 14-24	NORM
CC 25-35	DMHR
CC 36-46	DML$
CC 47-57	DOH$
CC 58-68	NDAY

D. Special Notes:

(1) The maximum number of aggregated data cards is 150. This capacity may be increased by changing the appropriate DIMENSION and DATA statements.

(2) This program may be modified to handle raw data cards by changing FORMAT statement number 9001.

```
C***********************************************************************
C
C
C        CLUSTER DATA PROGRAM - TO NORMALIZE AND STANDARDIZE DATA
C                                  FOR THE DISTANCE MATRIX PROGRAM
C
C              INPUT REQUIRED:
C              (1) FIRST DATA CARD MUST GIVE NUMBER OF FOLLOWING CARDS
C                  FORMAT (I3)
C                   ( AGGREGATED FORMAT DATA CARD ONLY )
C
C
C***********************************************************************
      DIMENSION IBOX(150),CBOX(150),JBOX(150),W(150),ABOX(750)
      DIMENSION TOT(5),AVG(5),SD(5),VMIN(5),VMAX(5),Y(5),X(4)
      REAL*8 CBOX
      INTEGER*2 JBOX
      DATA X/'MEAN','SDEV','MIN ','MAX '/
      DATA Y/'NORM','DMHR','DML$','DOH$','NDAY'/
      DATA W,NV,IER/150*1.0,5,0/
C
C   READ AND WRITE INPUT DATA THEN NORMALIZE
C
      READ (5,9000) NCARD
 9000 FORMAT (I3)
      WRITE (6,9500)
 9500 FORMAT ('1', 9X,'A/C',3X,'BUREAU',2X,'WORK',/,10X,'CODE',
     12X,'NUMBER',2X,'CODE',15X,'NORM',6X,'DMH',7X,'DM$',
     26X,'OVHD$',5X,'NDAYS',/)
      DO 1000 I=1,NCARD,1
      READ (5,9001) IBOX(I),CBOX(I),JBOX(I),A,B,C,D,E
 9001 FORMAT (I2,A8,A3,2F9.2,9X,2F9.2,9X,F9.2)
      WRITE (6,9501) I,IBOX(I),CBOX(I),JBOX(I),A,B,C,D,E
 9501 FORMAT (' ',I5,4X,I3,1X,A8,2X,A3,10X,5F10.3)
      IF (I.EQ.75) WRITE (6,9500)
      SQT=SQRT(A*A+B*B+C*C+D*D+E*E)
      ABOX(I)=A/SQT
      ABOX(I+NCARD)=B/SQT
      ABOX(I+2*NCARD)=C/SQT
      ABOX(I+3*NCARD)=D/SQT
      ABOX(I+4*NCARD)=E/SQT
 1000 CONTINUE
C
C   WRITE NORMALIZED DATA ONLY
C
      WRITE (6,9500)
      DO 1500 I=1,NCARD,1
      WRITE (6,9501) I,IBOX(I),CBOX(I),JBOX(I),ABOX(I),ABOX(I+NCARD),
```

```
     1ABOX(I+2*NCARD),ABOX(I+3*NCARD),ABOX(I+4*NCARD)
      IF (I.EQ.75) WRITE (6,9500)
 1500 CONTINUE
C
C    STANDARDIZE VECTOR THEN WRITE AND PUNCH
C
      CALL TALLY (ABOX,W,TOT,AVG,SD,VMIN,VMAX,NCARD,NV,IER)
      DO 2000 I=1,NCARD,1
      ABOX(I)=(ABOX(I)-AVG(1))/SD(1)
      DO 2000 J=1,4
      ABOX(I+J*NCARD)=(ABOX(I+J*NCARD)-AVG(J+1))/SD(J+1)
 2000 CONTINUE
      WRITE (6,9500)
      DO 3000 I=1,NCARD,1
      WRITE (6,9501) I,IBOX(I),CBOX(I),JBOX(I),ABOX(I),ABOX(I+NCARD),
     1ABOX(I+2*NCARD),ABOX(I+3*NCARD),ABOX(I+4*NCARD)
      WRITE (7,9503) IBOX(I),CBOX(I),JBOX(I),ABOX(I),ABOX(I+NCARD),
     1ABOX(I+2*NCARD),ABOX(I+3*NCARD),ABOX(I+4*NCARD)
 9503 FORMAT (I2,A8,A3,5F11.5)
      IF (I.EQ.75) WRITE (6,9500)
 3000 CONTINUE
C
C    WRITE STATISTICAL INFORMATION
C
      WRITE (6,9504) X
 9504 FORMAT ('1',///,11X,'STATISTICAL SUMMARY FOR NORMALIZED VECTOR ONL
     1Y',///,15X,2A10,A11,A10,//)
      DO 4000 I=1,5,1
      WRITE (6,9502) Y(I),AVG(I),SD(I),VMIN(I),VMAX(I)
 9502 FORMAT (' ',10X,A5,4F10.4,/)
 4000 CONTINUE
      STOP
      END


      SUBROUTINE TALLY(A,S,TOTAL,AVER,SD,VMIN,VMAX,NO,NV,IER)
      DIMENSION A(1),S(1),TOTAL(1),AVER(1),SD(1),VMIN(1),VMAX(1)
      IER=0
      DO 1 K=1,NV
      TOTAL(K)=0.0
      AVER(K)=0.0
      SD(K)=0.0
      VMIN(K)= 1.0E75
    1 VMAX(K)=-1.0E75
      SCNT=0.0
      DO 7 J=1,NO
      IJ=J-NO
      IF(S(J)) 2,7,2
```

```
  2 SCNT=SCNT+1.0
    DO 6 I=1,NV
    IJ=IJ+NO
    TOTAL(I)=TOTAL(I)+A(IJ)
    IF(A(IJ)-VMIN(I)) 3,4,4
  3 VMIN(I)=A(IJ)
  4 IF(A(IJ)-VMAX(I)) 6,6,5
  5 VMAX(I)=A(IJ)
  6 SD(I)=SD(I)+A(IJ)*A(IJ)
  7 CONTINUE
    IF (SCNT)8,8,9
  8 IER=1
    DO 85 I=1,NV
    VMIN(I)=-1.0E75
    VMAX(I)=1.0E75
    SD(I)=1.0E75
    AVER(I)=1.0E75
 85 CONTINUE
    GO TO 15
  9 DO 10 I=1,NV
 10 AVER(I)=TOTAL(I)/SCNT
    IF (SCNT-1.0) 13,11,13
 11 IER=2
    DO 12 I=1,NV
 12 SD(I)=0.0
    GO TO 15
 13 DO 14 I=1,NV
 14 SD(I)=SQRT(ABS((SD(I)-TOTAL(I)*TOTAL(I)/SCNT)/(SCNT-1.0)))
 15 RETURN
    END
```

APPENDIX C

Distance Matrix Program

A. Purpose: To provide a punched card deck of squared Euclidian distances between each pair of observation vectors (i.e. normalized and standardized input data cards). The distance matrix is a required input for the clustering program in Appendix D.

B. Inputs: A control card is required followed by a deck of normalized and standardized (N&S) cards. The formats are as follows:

Control Card: CC 1-3 number of N&S cards

N&S Deck: format is specified in Appendix B.

C. Outputs: The outputs are listings of input data and the squared Euclidian distance matrix. The distance matrix is stored in a linear array that corresponds to the lower half of the matrix. The listing and data cards of the distance matrix are printed and punched in the linear array format. Since a complicated procedure is required to locate the distance between any particular pair of observation vectors the format will not be specified in detail. Each distance matrix output card contains 10 distances. The total number of distances calculated and punched is:

$$\frac{n \times (n-1)}{2}$$ where n = number of N&S cards contained in the input deck.

D. Special Notes:

(1) The output deck must be kept in the <u>exact</u> order in which it is punched. The position of each number in the linear array determines which pair of observation vectors were used to compute that particular distance value.

(2) The maximum number of N&S input data cards in 200. This capacity may be increased by changing the DIMENSION statement.

```
C*****************************************************************************
C
C
C         COMPUTATION OF THE DISTANCE MATRIX FOR CLUSTERING PROGRAM
C         THE OUTPUT IS PUNCHED CARDS
C
C
C*****************************************************************************
      DIMENSION  BUR(200),X(5,200),BUF(10),NWRK(200),NACN(200)
      READ(5,17) N
17    FORMAT(I3)
      READ(5,27)(NACN(I), BUR(I),NWRK(I),(X(J,I),J=1,5),I=1,N)
27    FORMAT(I2,A8,A3,5F11.5,12X)
      WRITE(6,37)(I,NACN(I), BUR(I),NWRK(I),(X(J,I),J=1,5),I=1,N)
37    FORMAT(I6,I4,A10,A5,5F15.5)
      L=1
      DO 200 I=2,N
      II=I-1
      DO 200 J=1,II
      SUM=0.0
      DO 100 K=1,5
      DEL=X(K,I)-X(K,J)
100   SUM=SUM+DEL*DEL
      BUF(L)=SUM
      IF(L.EQ.10) GO TO 150
      L=L+1
      GO TO 200
150   WRITE(7 ,157) BUF
157   FORMAT(10F8.4)
      WRITE(6,167) BUF
167   FORMAT(10(2X,F8.4))
      L=1
200   CONTINUE
      IF(L.EQ.1) GO TO 300
      L=L-1
      WRITE(7 ,157)(BUF(M),M=1,L)
      WRITE(6,167) (BUF(M),M=1,L)
300   STOP
      END
```

APPENDIX D

Clustering Program

A. Purpose: The purpose of this program is to cluster together individual observations and groups of observations that are mathematically similar. Further discussion on clustering can be found in Ref. 8. The program begins by considering each input vector (observation) as a cluster with one observation. The two "closest" (based on Euclidian distance) clusters are combined into a single cluster and the distance from this new cluster to all other clusters is recomputed and the distance matrix is adjusted. The clustering continues until all observations are in a single cluster.

The program provides a detailed listing of the clusters that are combined at each step and at the users option the program will provide a detailed description of each cluster. A deck of punched cluster "descriptions" is produced at intervals specified by the user. A cluster "description" consists of the number of clusters, the number of observation vectors (i.e. number of elements) in each cluster and an ordered vector of observation numbers (an observation number is the number assigned to each observation vector based on its position when the distance matrix was produced). For example:

Suppose we have 6 observation vectors clustered as follows

Cluster no.	No. of observation vector
1	1,3
2	2
3	5,6,4

The cluster description would be

number of clusters = 3

number of observations in each cluster = 2,1,3

vector of observation numbers = 1,3,2,5,6,4,

B. Inputs: A control card and the distance matrix produced by Appendix C are required. The formats are as follows:

Control Card	CC 1-3	number of observation vectors
	CC 4-6	stage at which a complete cluster description begins in the output listing. (stage is the number of clusters currently existing in the program)
	CC 7-14	Beta value (See Ref. 8). The value -0.25000 used in all clustering done in this thesis.
	CC 15-17	KM1
	CC 18-20	KM2
	CC 21-23	KM3

$\left.\begin{array}{l}\text{KM1}\\ \text{KM2}\\ \text{KM3}\end{array}\right\}$ See below

Cluster "descriptions" are punched starting at Stage KM1 and continuing to Stage KM2 in increments of KM3. For example: if KM1 = 16

KM2 = 12

KM3 = 2

then cluster "descriptions" will be punched at stages 16, 14 and 12.

C. Outputs: A listing describing clustering as discussed in Part A is provided. A punched deck of cluster descriptions as specified by the control card is provided. The punched output format is as follows:

First card: CC 1-3 number of clusters

Second group of cards (as many as needed): number of observation vectors in each cluster with each card having a maximum of 20 numbers, each right each adjusted in a 3 column field.

Final group of cards (as many as needed): ordered vector of observation numbers as described in Part A. The format is the same as for the second group.

D. Special Notes:

(1) The maximum number of observation vectors is 100.

(2) The user must decide at what point further clustering ceases to be useful. The computer algorithm automatically proceeds until all observations are in a single cluster.

```
C***************************************************************************
C
C
C    CLUSTER ANALYSIS BY HEIRARCHICAL FLEXIBLE METHOD
C            WITH A PUNCH OPTION FOR CLUSTER DESCRIPTIONS
C
C
C***************************************************************************
      DIMENSION D(5000),NAME(100),NUMB(100),D1(100),D2(100),
     1 MCLUS(100),LOCN(101),NSTO(100)
      READ(5,17) N,NCLUS,BETA,KM1,KM2,KM3
17    FORMAT(2I3,F8.3,3I3)
      NX=N
      IF(N.LE.100) GO TO 50
      WRITE(6,27)
27    FORMAT('0TOO MANY ITEMS')
      STOP
50    NN=N*(N-1)/2
      ALPHA=(1.0-BETA)/2.0
      DO 51 I=1,N
      NAME(I)=I
      MCLUS(I)=I
      LOCN(I)=I
51    NUMB(I)=1
      LOCN(N+1)=N+1
      WRITE(6,57) N,ALPHA,BETA
57    FORMAT('1FLEXIBLE HIERARCHICAL CLUSTERING ALGORITHM',/,I4,
     1 ' ITEMS',7X,'ALPHA =',F6.3,7X,'BETA =',F7.3,//)
      READ(5,67) (D(K),K=1,NN)
67    FORMAT((10F8.0))
C
C    START NEXT ITERATION BY FINDING SMALLEST DISTANCE
C
70    DMIN=100000.0
      DO 100 I=2,N
      KK=(I-2)*(I-1)/2
      II=I-1
      DO 100 J=1,II
      K=KK+J
      IF(D(K).GE.DMIN) GO TO 100
      DMIN=D(K)
      IMIN=I
      JMIN=J
      KMIN=K
100   CONTINUE
      IMP1=IMIN+1
      IMM1=IMIN-1
      JMP1=JMIN+1
```

```
      JMM1=JMIN-1
      NM1=N-1
      IF(JMIN.EQ.IMIN-1)  GO TO 1045
      L=LOCN(JMIN+1)
      LL=LOCN(IMIN)-1
      DO 101  I=L,LL
101   NSTO(I)=MCLUS(I)
      N1=NUMB(IMIN)
      DO 102  I=1,N1
102   MCLUS(L+I-1)=MCLUS(LL+I)
      DO 103  I=L,LL
103   MCLUS(N1+I)=NSTO(I)
      DO 104  I=JMP1,IMM1
104   LOCN(I)=LOCN(I)+N1
1045  DO 105  I=IMIN,N
105   LOCN(I)=LOCN(I+1)
C
C     GROUPS IMIN AND JMIN WILL BE MERGED
C
      NUMB(JMIN)=NUMB(JMIN)+NUMB(IMIN)
      WRITE(6,107) N,NAME(JMIN),NAME(IMIN),NAME(JMIN),NUMB(JMIN)
107   FORMAT('OAT STAGE',I4,' MERGE GROUPS',I4,' AND',I4,' CALLING RESUL
     1TING GROUP',I4,' WITH',I4,' ELEMENTS')
C
C     EXTRACT D1 = DISTANCE VECTOR FOR JMIN GROUP
C             D2 = DISTANCE VECTOR FOR IMIN GROUP
C
      K=(JMIN-2)*(JMIN-1)/2
      IF(JMIN.EQ.1) GO TO 185
      DO 180  J=1,JMM1
180   D1(J)=D(K+J)
185   D1(JMIN)=0.0
      DO 190  I=JMP1,N
      K=((I-1)*(I-2)/2)+JMIN
190   D1(I)=D(K)
      K=(IMIN-2)*(IMIN-1)/2
      DO 200  J=1,IMM1
200   D2(J)=D(K+J)
      D2(IMIN)=0.0
      IF(IMIN.EQ.N) GO TO 215
      DO 210  I=IMP1,N
      K=((I-2)*(I-1)/2)+IMIN
210   D2(I)=D(K)
C
C     COMPUTE D3 = DISTANCE VECTOR FOR MERGED GROUP - PLACE IN D1
C
215   DO 220 I=1,N
220   D1(I)=ALPHA*(D1(I)+D2(I))+BETA*DMIN
```

```
C
C     INSERT D1  INTO MATRIX IN JMIN POSITION
C
      K=(JMIN-2)*(JMIN-1)/2
      IF(JMIN.EQ.1)  GO TO 235
      DO 230 J=1,JMM1
230   D(K+J)=D1(J)
235   DO 240 I=JMP1,N
      K=((I-2)*(I-1)/2)+JMIN
240   D(K)=D1(I)
C
C     REPACK MATRIX TO OMIT IMIN POSITION
C
      IF(IMIN.EQ.N) GO TO 280
      DO 250 I=IMIN,NM1
      KK=(I-2)*(I-1)/2
      KKK=KK+I-1
      DO 250 J=1,IMM1
250   D(KK+J)=D(KKK+J)
      IF(IMIN.EQ.N-1) GO TO 265
      DO 260 I=IMP1,NM1
      KK=(I-2)*(I-1)/2
      KKK=KK+I
      II=I-1
      DO 260 J=IMIN,II
260   D(KK+J)=D(KKK+J)
C
C     PACK NAME AND NUMBER LISTS
C
265   DO 270 I=IMIN,NM1
      NAME(I)=NAME(I+1)
270   NUMB(I)=NUMB(I+1)
280   N=N-1
      IF (KM1.LT.KM3) GO TO 281
      IF (N.EQ.KM1) GO TO 300
281   IF(N.GT.NCLUS) GO TO 70
      WRITE(6,287) N
287   FORMAT('OSTAGE',I4)
      DO 290 I=1,N
      L=LOCN(I)
      LL=LOCN(I+1)-1
290   WRITE(6,297) NAME(I),(MCLUS(K),K=L,LL)
297   FORMAT(' GROUP',I4,I10,19I4,/,(16X,20I4))
      IF(N.GT.1) GO TO 70
      STOP
C
C     PUNCH CLUSTER DESCRIPTIONS
C     STARTING AT KM1 ENDING AT KM3 IN INCREMENTS OF KM2
```

```
C
  300 WRITE (7,301) N
301    FORMAT (I3)
       WRITE (7,302)  (NUMB(I),I=1,N)
302    FORMAT (20I3,20X)
       WRITE (7,302)  (MCLUS(I),I=1,NX)
       KM1=KM1-KM2
       GO TO 281
       END
```

APPENDIX E

Cluster Reallocation Program

A. Purpose: This program ensures that each observation vector is assigned to the closest group centroid calculated from a cluster description supplied by the clustering Program in Appendix D. The program takes the normalized and standardized data cards and based on a specified cluster description a group centroid is calculated for each cluster. Each observation vector is then assigned to the closest centroid based on Euclidian distance. If any observation vector moves to a different cluster from its present assignment then the group centroids are recalculated and the observation vectors are again assigned to the closest group centroid. This reallocation process continues until a stable configuration is reached (i.e. no observation vector shifts clusters).

B. Inputs: A control card is followed by the normalized and standardized data deck then the cluster description(s). The formats are as follows:

Control card: CC 1-3 number of N&S data cards

N&S Data Deck: format is the same as specified in Appendix B.

Cluster Descriptions: format is the same as specified in Appendix D.

C. Outputs: The normalized and standardized data is printed along with the initial cluster description. The program

then prints a series of cluster centroids and cluster descriptions until a stable configuration is reached. Then a deck of reallocated cluster descriptions is punched. The reallocated cluster descriptions have the same format as specified in Appendix D.

D. Special Notes:

(1) The program is set to handle multiple cluster descriptions. Place any additional cluster descriptions behind the first one.

(2) The maximum number of N&S data cards is 100. The DIMENSION statement may be changed to accomodate larger capacity.

(3) The maximum number of clusters that can be specified in a cluster description is 50. This capacity may be changed in the DIMENSION statement.

(4) Normally about 5% or less of the observation vectors will change clusters. If the percentage exceeds this limit then an input error or a very unstable cluster configuration may have caused the problem. The results should be carefully checked before continuing.

```
C***************************************************************************
C
C
C     K MEAN CLUSTER REALLOCATION SYSTEM
C       OUTPUT IS A PUNCHED DECK OF REALLOCATED CLUSTERS
C
C
C***************************************************************************
      DIMENSION X(5,100),MCLUS(100),LOCN(101),NAME(100),NUMB(100),
     1  CENTR(5,50),ID(100)
      WRITE (6,9500)
C
C    READ INPUT DATA
C
      READ(5,7) N
7     FORMAT(I3)
      READ(5,17)((X(J,I),J=1,5),I=1,N)
17    FORMAT(13X,5F11.5,12X)
20    READ (5,7,END=9999) NCLUS
      READ(5,37)(NUMB(I),I=1,NCLUS)
37    FORMAT(20I3,20X)
      READ(5,37)(MCLUS(I),I=1,N)
      LOCN(1)=1
      DO 50 K=1,NCLUS
      LOCN(K+1)=LOCN(K)+NUMB(K)
50    NAME(K)=MCLUS(LOCN(K))
      DO 60 K=1,NCLUS
      NK=NUMB(K)
      L=LOCN(K)-1
      DO 60 I=1,NK
      MC=MCLUS(L+I)
60    ID(MC)=K
C
C    WRITE CLUSTER
C
70    WRITE(6,287) NCLUS
287   FORMAT('0STAGE',I4)
      DO 290 I=1,NCLUS
      L=LOCN(I)
      LL=LOCN(I+1)-1
290   WRITE(6,297)I,NAME(I),(MCLUS(K),K=L,LL)
297   FORMAT(I5,' GROUP',I4,I10,19I4,/,(16X,20I4))
C
C    COMPUTE GROUP CENTROIDS
C
      DO 100 K=1,NCLUS
      DO 100 J=1,5
100   CENTR(J,K)=0.0
```

```
      DO 150 K=1,NCLUS
      NK=NUMB(K)
      ZNK=NK
      L=LOCN(K)-1
      DO 140 I=1,NK
      MC=MCLUS(L+I)
      DO 140 J=1,5
  140 CENTR(J,K)=CENTR(J,K)+X(J,MC)
      DO 145 J=1,5
  145 CENTR(J,K)=CENTR(J,K)/ZNK
  150 CONTINUE
      WRITE(6,147)
  147 FORMAT(' ')
      WRITE(6,157)(K,NAME(K),(CENTR(J,K),J=1,5),K=1,NCLUS)
  157 FORMAT(I5,' GROUP',I4,' CENTROID',5F10.3)
      WRITE(6,147)
C
C    ASSIGN EACH POINT TO NEAREST CENTROID
C
      ISW=0
      DO 700 I=1,N
      DMIN=10000.0
      DO 600 K=1,NCLUS
      D=0.0
      DO 550 J=1,5
      DXJ=X(J,I)-CENTR(J,K)
  550 D=D+DXJ*DXJ
      IF(D.GE.DMIN) GO TO 600
      DMIN=D
      KMIN=K
  600 CONTINUE
      IF(ID(I).NE.KMIN) ISW=1
  700 ID(I)=KMIN
      IF(ISW.EQ.0)  GO TO 1000
C
C    DEFINE NEW CLUSTERING
C
      KSKIP=0
      LOCN(1)=1
      DO 850 K=1,NCLUS
      KK=K-KSKIP
      NUMB(KK)=0
      L=LOCN(KK)-1
      DO 800 I=1,N
      IF(ID(I).NE.K) GO TO 800
      NUMB(KK)=NUMB(KK)+1
      NN=L+NUMB(KK)
      MCLUS(NN)=I
```

```
      ID(I)=KK
800   CONTINUE
      IF(NUMB(KK).EQ.0)  GO TO 820
      LOCN(KK+1)=LOCN(KK)+NUMB(KK)
      GO TO 850
820   KSKIP=KSKIP+1
      WRITE(6,827) K,NAME(K)
827   FORMAT(I5,' GROUP',I4,' HAS DISAPPEARED')
850   CONTINUE
      NCLUS=NCLUS-KSKIP
      DO 900 K=1,NCLUS
      L=LOCN(K)
900   NAME(K)=MCLUS(L)
C
C    GO BACK AND SEE IF ANY OTHER POINTS WANT TO SHIFT CLUSTERS
C
      GO TO 70
C
C    PUNCH REALLOCATED CLUSTER DESCRIPTION
C
1000  WRITE (6,9500)
9500  FORMAT ('1',//)
      WRITE (7,9501) NCLUS
9501  FORMAT (I3)
      WRITE (7,9502) (NUMB(I),I=1,NCLUS)
9502  FORMAT (20I3,20X)
      WRITE (7,9502) (MCLUS(I),I=1,N)
C
C    GO BACK AND REALLOCATE ANY REMAINING CLUSTER DESCRIPTIONS
C
      GO TO 20
9999  STOP
      END
```

APPENDIX F

Aggregation Program

A. Purpose: This program combines groups of raw data cards into aggregated data cards. A listing is provided for the new aggregated data. The aggregation procedure consists of calculating a numerical average for each variable (excluding airframe change man-hours on aircraft raw data cards) on the raw data card. Two additional fields are calculated and punched into the new cards: NDAY and NOBS (number of raw data cards that are combined to produce the aggregated data card). These aggregated data cards are used as input requirements for programs in Appendices B, G, K and L.

B. Inputs: A series of control cards are required. Each control card <u>must</u> be the first card of the group to be aggregated. The control card is punched in CC 1-3 with the exact number of raw data cards following it. The raw data cards are in the format specified in Appendix A. The input deck as described is called the preaggregation deck. It is used as an input deck for the program in Appendix J. Once this deck is created it should be kept in addition to the aggregated data cards. (Note: the control cards are referred to as I3 separator cards in the program)

C. Outputs: The output consists of a reformatted raw data card with two additional fields, NDAY and NOBS. These new cards are referred to as aggregated data cards. (Each aggregated data card is an estimation of a NARF process.) The program provides a listing of the aggregated data cards. The aggregated data card format follows:

CC 1-2	Aircraft/engine type
CC 3	blank
CC 4-10	identification code
CC 11	blank
CC 12-13	work type
CC 14-22	NORM
CC 23-31	DMHR
CC 32-40	DLB$
CC 41-49	DML$
CC 50-58	DOH$
CC 59-67	NIFR
CC 68-76	NDAYS
CC 77-80	NOBS

D. Special Notes:

(1) The maximum number of control cards is 150. This capacity may be increased by the change of the DIMENSION and DATA statements of the program.

(2) The first 13 card columns of the first raw data card will become an "identifier" and are transferred to the aggregated data card.

```
C***********************************************************************
C
C
C         AGGREGATION PROGRAM FOR RAW DATA ( ENGINE OR AIRCRAFT )
C
C                INPUT DATA:
C                     (1) RAW DATA CARDS WITH I3 SEPARATOR CARDS
C
C
C***********************************************************************
      DIMENSION A(150),B(150),C(150),D(150),E(150),F(150),G(150)
      DIMENSION IB(150),AB(150),KB(150),IH(150),S(8)
      REAL*8 AB
      INTEGER*2 IB
      DATA A,B,C,D,E,F,G/1050*0.0/
      DATA S/'NORM','DMHR','DLB$','DML$','DOH$','NIFR','NDAY','NOBS'/
      WRITE (6,9500) S
 9500 FORMAT ('1',//,17X,'IDENTIFICATION',A10,4A12,1X,2A12,A10,//)
      I=0
 1000 I=I+1
C
C   READ INPUT DATA FOR GROUP TO BE AGGREGATED
C
      READ (5,9000,END=9999) NN
 9000 FORMAT (I3)
      IF (NN.EQ.1) GO TO 6000
C
C   FIRST CARD WILL CONTAIN IDENTIFICATION FOR THE GROUP
C
      READ (5,9001) IB(I),AB(I),KB(I),ID,JD,AX,BX,CX,DX,EX,FX
 9001 FORMAT (I2,A8,A3,2I5,F6.0,6X,F6.0,3F7.0,F6.2)
      A(I)=A(I)+AX
      B(I)=B(I)+BX
      C(I)=C(I)+CX
      D(I)=D(I)+DX
      E(I)=E(I)+EX
      F(I)=F(I)+FX
      NDAYS=JD-ID
      IF (NDAYS.GE.635) NDAYS=NDAYS-635
      G(I)=G(I)+FLOAT(NDAYS)
      IH(I)=NN
C
C   NOW READ THE REST OF GROUP AND COMPUTE AVERAGES
C
      DO 1500 J=2,NN,1
      READ (5,9002) ID,JD,AX,BX,CX,DX,EX,FX
 9002 FORMAT (13X,2I5,F6.0,6X,F6.0,3F7.0,F6.2)
      A(I)=A(I)+AX
```

```
      B(I)=B(I)+BX
      C(I)=C(I)+CX
      D(I)=D(I)+DX
      E(I)=E(I)+EX
      F(I)=F(I)+FX
      NDAYS=JD-ID
      IF (NDAYS.GE.635) NDAYS=NDAYS-635
      G(I)=G(I)+FLOAT(NDAYS)
 1500 CONTINUE
      A(I)=A(I)/FLOAT(NN)
      B(I)=B(I)/FLOAT(NN)
      C(I)=C(I)/FLOAT(NN)
      D(I)=D(I)/FLOAT(NN)
      E(I)=E(I)/FLOAT(NN)
      F(I)=F(I)/FLOAT(NN)
      G(I)=G(I)/FLOAT(NN)
C
C   GO BACK FOR NEXT GROUP TO BE AGGREGATED
C
      GO TO 1000
C
C   LOOP FOR SINGLE CARD GROUPS
C
 6000 READ (5,9001) IB(I),AB(I),KB(I),ID,JD,A(I),B(I),C(I),D(I),E(I),
     1F(I)
      NDAYS=JD-ID
      IF (NDAYS.GE.635) NDAYS=NDAYS-635
      G(I)=FLOAT(NDAYS)
      IH(I)=NN
C
C   GO BACK FOR NEXT GROUP TO BE AGGREGATED
C
      GO TO 1000
C
C   PRINT AND PUNCH AGGREGATED DATA CARDS
C
 9999 LL=I-1
      DO 2000 J=1,LL,1
      IF (J.EQ.71) WRITE (6,9500) S
      WRITE (6,9501) J,IB(J),AB(J),KB(J),A(J),B(J),C(J),D(J),E(J),
     1F(J),G(J),IH(J)
 9501 FORMAT (' ',10X,I2,5X,I2,A8,A3,7F12.2,I8)
      WRITE (7,9502) IB(J),AB(J),KB(J),A(J),B(J),C(J),D(J),E(J),
     1F(J),G(J),IH(J)
 9502 FORMAT (I2,A8,A3,7F9.2,I4)
 2000 CONTINUE
      STOP
      END
```

APPENDIX G

Dominance Program for Aggregated Data

A. Purpose: The purpose of this program is to take aggregated data cards as an input and then display in a matrix the dominance relationships between the data cards. Recall that each aggregated data card represents an estimated process for the NARF. The specified input/output variables (in Chapter 2) are read from the input deck and then each process is converted to a unit output. The dominance matrix is printed and can be inspected for dominance among processes with the <u>same</u> output. The procedure to eliminate the dominance is described in Chapter 2.

B. Inputs: The program requires a series of control cards to be inserted among the aggregated data cards. The control cards are punched with the number of aggregated data card following them (and prior to the next control card) in card columns 1-3. These control cards cause the aggregated data cards to be read in small groups thereby reducing the size of the dominance matrix. The control card should contain a number between 10 and 20 for most groups. For example, if we have 72 aggregated data cards then six control cards with 12 punched in CC 2-3 would suffice. However, the user must ensure that <u>all processes</u> with the <u>same output</u> are contained in the same control

card group. The dominance matrix is only calculated for those processes in the same group.

C. Outputs: The output consists of a listing of the aggregated data group (input/output variables only) and a listing of the unit output processes associated with each data card in the group. The dominance matrix is printed with the above listings. Each new group is started on a new page for ease of reference. The dominance matrix consists of three symbols defined as follows:

(1) - means there is no dominance relationship between process A and process B (A and B are defined below)

(2) X means that process A is dominated by (i.e. strictly greater than) process B

(3) O means that process A dominates (i.e. strictly less than) process B.

Each symbol is in a particular row and column of the dominance matrix. (the column numbers are the same as the row numbers since the dominance matrix is always a square matrix.) "A" is the row number of the symbol and "B" is the column number of the symbol.

D. Special Notes: None.

```
C***************************************************************************
C
C
C    CHECK DOMINANCE ON RAW AGGREGATED DATA
C
C
C***************************************************************************
      DIMENSION X(5,100), IOUT(100),IB(25),JB(25),KB(25),NB(25),Y(6)
      REAL*8 JB
      DATA Y/'NORM','DMHR','DML$','DOH$','NDAY','NOBS'/
      DATA IDASH/'-'/,IOH/'O'/,ISTR/'*'/,IXX/'X'/
      L=1
C
C    READ INPUT DATA
C
5     READ(5,7,END=9999) N
7     FORMAT(I3)
      READ (5,17)((IB(I),JB(I),KB(I),(X(J,I),J=1,5),NB(I)),I=1,N)
17    FORMAT (A2,A8,A3,2F9.2,9X,2F9.2,9X,F9.2,I4)
      STRL=L
C
C    WRITE INPUT DATA
C
      WRITE (6,50) Y
50    FORMAT ('1',/,6X,'IDENTIFICATION',8X,A4,4A15,A11,/)
90    DO 99 I=1,N
      WRITE (6,97) L,IB(I),JB(I),KB(I),(X(J,I),J=1,5),NB(I)
97    FORMAT(' ',I4,2X,A2,A8,A3,5F15.3,I8)
      L=L+1
99    CONTINUE
C
C    NORMALIZE INPUT DATA
C
      DO 200 I=1,N
      Z=X(1,I)
      DO 150 J=1,5
150   X(J,I)=X(J,I)/Z
200   CONTINUE
      WRITE(6,250)
250   FORMAT(' ')
      L=STRL
C
C    WRITE NORMALIZED VECTOR
C
      DO 199 I=1,N
      WRITE(6,197) L,(X(J,I),J=1,5)
197   FORMAT (' ',I4,15X,5F15.5)
      L=L+1
```

```
199   CONTINUE
      WRITE(6,1047)
      L=STRL
C
C     COMPUTE AND WRITE THE DOMINANCE MATRIX
C
      DO 600 I=1,N
      DO 500 J=1,N
      IOUT(J)=IDASH
      IF(I.EQ.J) GO TO 490
      IF((X(2,J).LE.X(2,I)).AND.(X(3,J).LE.X(3,I)).AND.(X(4,J).LE.X(4,I)
     - ).AND.(X(5,J).LE.X(5,I)))   IOUT(J)=IOH
      IF((X(2,J).GE.X(2,I)).AND.(X(3,J).GE.X(3,I)).AND.(X(4,J).GE.X(4,I)
     - ).AND.(X(5,J).GE.X(5,I)))   IOUT(J)=IXX
      GO TO 500
490   IOUT(J)=ISTR
500   CONTINUE
      WRITE(6,507) L,(IOUT(K),K=1,N)
507   FORMAT(I6,2X,100A2)
      L=L+1
600   CONTINUE
C
C     GO BACK FOR ANOTHER GROUP OF DATA CARDS
C
      GO TO 5
9999  STOP
      END
```

APPENDIX H

Listing of Aggregated Aircraft Data

The following listing provides the values of the aggregated data cards used in the aircraft model. The columns are as follows:

(1) aircraft code

(2) identification code

(3) work code

(4) NORM

(5) DMHR

(6) DLB$

(7) DML$

(8) DOH$

(9) NIFR

(10) NDAY

(11) NOBS

The format of these cards is described in Appendix F.

LISTING OF AGGREGATED DATA FOR AIRCRAFT

10	155124	41	8300.00	8546.00	52253.00	17210.00	53745.00	14.42	90.00	1
10	152794	42	9300.00	9221.00	58201.00	26476.00	56230.50	15.21	96.50	2
11	152482	32	10466.66	10776.00	66327.31	29337.66	73043.00	15.68	106.00	3
11	150531	33	11847.06	12893.70	78238.25	33550.12	86277.75	15.36	109.76	17
11	151708	34	11620.00	13001.60	77663.38	29554.80	77302.00	14.29	105.40	5
21	155810	11	15000.00	15418.00	93390.00	74677.00	98584.00	17.29	172.00	1
21	155752	21	7777.02	8278.75	51621.68	21877.52	54436.41	15.48	67.82	44
21	153783	22	8200.00	8849.50	56232.25	21124.75	55686.50	15.04	61.25	4
21	153910	41	12401.86	12690.57	77723.19	17800.57	79728.81	13.82	113.76	21
21	153801	62	10600.00	10998.00	68624.00	22934.00	68747.00	14.58	85.00	1
22	152214	13	23500.00	23439.00	149150.00	56390.00	139378.00	14.72	256.00	1
22	150482	14	21000.00	21293.00	131024.00	62424.00	133386.00	15.35	244.00	1
22	153023	22	10500.00	11116.00	71161.00	20473.00	66952.00	14.27	70.00	1
22	152279	23	11228.57	10989.86	68978.13	26516.00	68350.69	14.89	93.86	7
22	149431	24	10900.00	11050.00	69056.00	26891.00	68163.25	14.85	98.25	4
22	148407	25	11250.00	11542.00	71773.00	26468.75	72956.25	14.84	110.25	4
22	153065	42	12187.50	12152.75	75018.13	20091.13	87121.38	14.05	91.88	8
22	152229	43	17731.71	18252.57	113446.69	34615.86	116201.25	14.47	145.14	7
22	151442	44	13800.00	14514.00	90096.50	28120.50	90553.00	14.40	125.00	2
22	149421	45	13916.66	15414.50	95536.00	27592.66	96896.50	14.26	138.83	6
22	153026	52	14000.00	14287.00	93748.00	34207.00	92081.50	15.42	133.50	2
22	151464	53	17587.40	18046.40	112028.00	35541.00	114137.00	14.51	144.80	5
22	150468	54	19420.00	20087.80	121021.75	31264.80	122414.75	13.74	156.20	5
22	151000	55	16300.00	18059.50	114997.50	37077.25	110355.00	14.54	151.50	4
22	152291	63	12000.00	10377.00	66285.00	24097.00	61774.00	14.66	93.00	1
22	151431	64	11500.00	11757.50	75436.00	26768.00	72011.50	14.80	91.50	2
22	149458	65	12500.00	14855.66	94877.31	29671.33	90486.63	14.47	139.67	3
22	148369	66	12400.00	12461.00	78940.00	34868.00	78376.00	15.42	92.00	1
23	150481	34	14800.00	11591.00	69925.00	18356.00	78419.00	14.38	92.00	1
25	150863	11	8850.00	8483.50	50988.50	21270.50	57653.00	15.32	128.50	2
25	150329	21	6795.23	6919.62	43129.66	17467.29	44138.00	15.15	72.43	21
25	150347	22	6600.00	5732.00	34853.00	18924.00	33949.00	15.30	65.00	1
25	149180	61	7955.55	8676.78	55170.55	23124.78	51984.33	15.01	93.00	9
26	147043	11	9950.00	10230.00	62253.00	24488.50	66432.00	14.97	155.50	2
26	147904	12	9800.00	9591.00	58216.00	21173.00	66514.00	15.21	142.00	1
26	148659	21	7066.66	6662.78	40555.33	16228.66	44624.66	15.23	69.22	9
26	147902	61	8300.00	8245.00	49593.00	21101.00	58087.00	15.62	84.00	1
26	148698	62	7900.00	9650.00	62212.00	31065.00	59107.00	15.79	90.00	1
27	145608	12	9100.00	9337.00	57724.00	20860.00	58037.00	14.63	166.00	1
27	146856	22	8400.00	7073.50	44530.00	14480.00	49572.00	15.42	73.00	2
27	146897	23	7150.00	8687.00	54783.00	10784.50	53867.00	13.76	72.00	2
27	146858	61	7300.00	7818.00	50012.00	20353.00	47482.00	15.07	70.00	1
27	146835	63	7828.57	9427.00	60481.43	19554.57	56417.71	14.49	89.71	7

31	6212576	20	5125.00	4895.75	29267.75	14474.00	32846.00	15.65	56.25	4
32	148040	22	4500.00	5188.00	30286.00	21813.00	38578.00	17.48	66.00	1
32	147144	23	4200.00	4788.00	29705.00	12382.00	30289.00	15.12	59.00	1
32	147146	24	4500.00	3423.00	20322.00	16512.00	25153.00	18.11	49.00	1
33	149704	21	4400.00	3394.00	21190.00	11171.00	21113.00	15.76	60.00	1
33	152123	23	4400.00	3820.00	22012.00	7478.00	28059.00	15.07	53.00	1
34	151554	33	2850.00	3975.00	24283.00	7062.00	25793.50	14.23	53.00	2
34	149905	34	4981.25	5081.13	30939.38	10924.63	32421.38	14.56	62.06	16
34	149000	35	5237.50	4675.83	28984.66	11055.08	29306.25	14.84	61.83	24
34	149683	36	4000.00	4227.33	26139.33	9328.66	26939.66	14.51	55.33	3
35	154107	12	10100.00	9577.00	57499.00	35159.00	66819.00	16.65	190.00	1
35	152701	22	4200.00	4041.00	23549.80	9735.40	27086.60	14.93	54.20	5
41	152530	22	4500.00	4593.00	28356.00	17415.00	28774.00	16.23	59.00	1
41	150962	23	4400.00	4313.00	26762.00	14927.00	26769.16	15.88	60.00	6
41	150963	43	5200.00	4805.00	27975.00	9598.00	34929.00	15.09	69.00	1
42	153958	22	4000.00	4051.64	25045.91	12987.91	25452.45	15.69	55.00	11
42	153975	42	4500.00	4684.25	27746.25	10510.25	31691.25	15.01	61.00	4
43	156429	21	3677.78	3150.81	19155.07	11228.52	20692.22	16.25	49.89	27
44	152495	23	4500.00	4291.00	29007.00	15100.00	29174.00	17.08	54.00	1
45	153412	22	4000.00	3554.00	22186.50	16216.50	24443.50	17.67	58.50	2
48	154882	21	4900.00	5250.50	32378.50	27517.00	37008.00	18.55	64.00	2
48	152402	22	4900.00	5300.20	31724.60	22330.00	36990.20	17.13	69.00	5
48	154876	71	7800.00	6931.71	40263.43	36441.14	49491.71	18.10	65.14	7
48	151698	72	7100.00	6793.11	41754.78	42036.11	43686.22	18.91	65.44	9
48	153287	82	5500.00	5692.66	35056.33	24997.33	33674.66	16.47	63.67	3
49	156654	21	5500.00	5298.00	33481.00	17220.00	32757.00	15.75	53.00	1
49	156656	71	7000.00	7128.00	43771.00	21951.00	46631.00	15.76	62.00	1

APPENDIX I

Listing of Aggregated Engine Data

The following listing provides the values of the aggregated data cards used in the engine model. The columns are as follows:

(1) engine code

(2) identification code

(3) work code

(4) NORM

(5) DMHR

(6) DLB$

(7) DML$

(8) DOH$

(9) NIFR

(10) NDAY

(11) NOBS

The format of these cards is described in Appendix F.

LISTING OF AGGREGATED DATA FOR ENGINES

51	2AN0101	01	435.00	396.00	2377.50	5015.50	3013.00	26.83	33.50	2
51	2BN0301	04	260.00	239.50	1447.50	599.00	1735.50	15.79	24.00	2
53	2JN0301	01	200.00	168.00	1010.40	1175.00	1349.40	21.08	31.00	5
53	2HN0124	01	444.29	417.29	2498.68	4656.61	3168.11	24.77	35.29	28
53	2JN0306	04	260.00	222.57	1324.43	2115.00	1596.57	23.43	27.86	7
55	2PP0102	01	470.00	562.00	3240.00	4950.00	4255.00	22.14	48.00	1
55	2PP0301	04	260.00	198.00	1203.93	1179.00	1477.07	19.60	23.07	14
56	25L0301	04	70.00	42.20	279.60	1433.20	246.80	62.92	34.80	5
56	28L0301	03	40.00	48.00	301.00	1270.00	253.00	38.28	19.00	2
57	2TP0301	01	160.00	194.00	1169.00	3475.00	1603.00	32.20	30.00	1
63	2JT0101	01	697.00	780.50	4559.00	12106.50	6212.00	28.85	53.50	2
63	2KT0302	04	371.00	425.00	2627.00	1122.00	3084.00	16.08	28.00	1
65	24T0104	01	700.00	697.10	4121.20	10781.80	5609.80	29.38	47.20	10
65	22T0101	01	700.00	689.50	4220.00	18100.50	5121.50	39.88	37.00	4
65	24T0101	01	700.00	746.00	4562.00	14577.50	5429.00	32.94	36.50	2
65	25T0301	03	240.48	251.81	1517.90	3761.71	2020.14	29.72	31.14	21
65	26T0301	03	233.64	300.91	1891.82	3847.18	2242.73	27.05	24.91	11
65	27T0302	03	245.00	300.50	1872.00	4555.66	2223.50	28.74	22.83	6
65	28T0318	04	284.29	329.07	1991.43	5488.86	2436.64	29.11	31.43	14
65	25T0303	04	300.53	272.16	1634.42	4803.31	2172.42	32.10	38.11	19
65	26T0303	04	286.67	297.11	1874.83	5895.05	2200.61	35.80	31.89	18
65	28T0301	04	280.00	326.00	2197.00	2530.75	2433.25	22.20	24.50	4
66	2EB0301	04	280.83	346.17	2116.75	4513.08	2546.25	26.52	36.08	12
67	2DW0103	01	723.00	614.75	3773.75	23611.75	4794.75	54.51	48.25	4
67	2KW0301	04	370.91	245.82	1540.73	7556.27	1875.36	42.16	31.18	11
69	24W0102	01	688.57	598.29	3671.50	30026.14	4553.29	64.28	46.50	14
69	28W0301	03	280.00	228.50	1371.50	4827.00	1706.50	34.60	27.50	2
69	25W0301	04	273.85	290.54	1771.92	16476.15	2224.92	68.84	38.62	13
72	25B0301	04	525.00	627.00	3825.50	18771.00	5112.00	44.98	41.50	2
74	25E0301	03	200.00	214.86	1329.86	4793.71	1676.14	38.79	29.29	7
74	25E0303	04	400.00	463.50	2774.00	6593.13	3621.50	28.19	32.75	8
73	25D0307	03	320.00	362.67	2137.17	1907.00	2943.83	19.97	25.83	6
75	26D0309	03	349.09	253.64	1595.27	4271.18	1892.73	31.42	27.55	11
75	28D0303	03	320.00	221.67	1374.00	3935.00	1573.33	31.14	21.00	3
73	28D0334	04	468.79	535.00	3173.15	4557.63	4156.51	22.98	29.09	33
73	26D0301	04	498.18	460.64	2783.68	3684.77	3380.77	22.21	35.00	22
75	27D0302	04	466.67	421.33	2636.56	5532.33	3222.33	29.22	37.44	9
75	28D0301	04	460.91	432.45	2711.45	6988.63	2941.09	30.03	33.00	11
81	21R0101	01	404.00	420.80	2548.20	6012.80	3414.40	27.76	37.80	5
81	28N0324	03	161.43	129.00	790.57	1304.71	1033.00	24.92	14.71	7
81	26R0301	04	220.00	128.17	779.33	1794.00	899.33	27.61	18.33	6
82	25N0301	03	100.00	160.00	947.11	504.22	1301.67	17.63	13.67	9
83	24N0205	02	416.15	463.77	2697.77	6199.92	3548.92	26.95	43.08	13

83	21N0201	02	400.00	435.09	2506.91	4047.73	3426.55	23.03	35.09	11
83	22N0201	02	434.69	414.53	2527.38	7898.84	3068.16	32.86	32.78	32
83	28N0211	04	260.00	281.20	1650.40	2000.00	2104.60	20.09	34.80	5
84	2GJ0101	01	431.82	421.09	2491.73	6027.18	3234.27	28.01	36.45	11
84	2GF0301	04	220.00	206.33	1215.89	1175.00	1554.00	18.95	27.00	9
85	2AR0101	01	450.00	442.20	2639.80	4764.24	3438.44	24.32	37.96	25
85	2CR0101	01	434.00	396.40	2447.60	6075.10	3100.10	29.27	32.90	10
86	2BR0201	02	470.00	427.50	2645.00	8568.00	3128.00	33.83	29.50	2
86	25N0201	05	320.00	307.07	1816.40	1722.23	2391.07	19.62	34.97	30
86	26N0201	05	320.00	288.86	1757.14	2849.29	2114.97	23.25	34.23	35
86	27N0201	05	305.00	278.19	1715.58	2650.33	2125.65	23.32	28.19	48
86	28N0701	05	280.00	259.42	1581.18	4096.71	1750.61	29.07	26.71	38
89	24P0106	01	489.00	455.30	2667.85	6881.35	3502.80	28.86	38.00	20
89	23P0101	01	440.00	424.40	2588.80	7299.00	3213.80	31.02	32.40	5
89	24P0101	01	440.00	388.83	2364.83	9444.50	2747.00	37.59	30.83	6
89	25P0304	03	157.69	172.35	1024.27	1341.08	1392.19	22.94	20.73	26
89	26P0301	03	150.53	168.74	1014.74	1476.89	1204.79	22.31	16.58	19
89	27P0302	03	160.00	180.38	1108.59	2043.79	1283.31	25.43	17.55	29
89	28P0330	04	193.53	222.53	1349.41	1893.65	1685.71	22.55	28.74	34
89	27P0301	04	210.00	226.74	1408.05	2211.53	1699.95	24.53	22.68	19
91	25J0311	03	281.84	316.14	1941.46	2062.11	2381.67	20.84	28.41	98
91	28J0345	04	570.93	609.10	3694.37	3983.17	4664.30	20.73	36.49	107
91	28J0304	04	579.73	536.81	3318.11	5335.38	3772.16	23.90	32.19	37
92	24J0125	01	1260.00	1270.69	7576.30	11865.92	10212.84	22.88	50.08	26
92	21J0101	01	1310.70	1324.38	7794.63	13386.49	10300.46	23.91	41.45	71
92	23J0101	01	1322.58	1293.26	7978.13	15248.19	10095.32	25.85	38.87	31
92	24J0101	01	1280.00	1251.53	7721.12	13375.23	8649.64	23.94	39.94	17
93	2EU0301	04	580.00	406.50	2720.00	730.00	3234.50	16.60	28.00	4
95	24K0111	01	1260.00	1216.38	7090.88	11050.88	9428.50	22.92	47.00	8
95	21K0101	01	1240.95	1144.76	6740.66	11362.47	9386.23	24.10	45.52	21
95	22K0101	01	1207.14	1220.64	7152.57	9423.93	9123.50	21.02	40.64	14
95	23K0101	01	1216.07	1248.82	7603.36	12323.93	9669.07	23.73	40.36	28
95	24K0101	01	1249.30	1214.23	7458.59	12699.07	8611.24	23.71	39.53	57
95	25K0301	03	314.31	318.81	1926.33	1875.76	2455.88	20.51	25.79	42
95	27K0302	03	337.14	332.71	2051.51	2193.31	2447.31	21.09	28.80	35
95	28K0339	04	395.00	420.83	2473.50	2671.46	3213.63	18.45	37.46	24
95	25K0325	04	450.00	579.41	3464.05	2856.95	4500.32	18.90	34.09	22
95	26K0301	04	524.38	513.83	3169.25	3705.29	3743.95	21.23	32.25	89

APPENDIX J

Average Cost of Labor Program

A. Purpose: This program computes the average cost of labor in dollars per man-hour for each aggregation group (i.e. the raw data cards being aggregated). A weighted average labor cost is then computed based on the number of observations in each group. This weighted average is used in the Linear Economic Model to convert man-hours into a dollar cost in the objective function.

B. Inputs: The input data for this program is the preaggregation deck described as an input to the Aggregation Program in Appendix F.

C. Outputs: A listing of the average cost of labor for each aggregation group followed by the weighted average cost of labor for all the groups.

D. Special Notes: None.

```
C***********************************************************************
C
C        CALCULATION OF THE AVERAGE COST OF LABOR PER HOUR
C
C***********************************************************************
      REAL*8 AB
      WRITE (6,9400)
9400  FORMAT ('1',4X,'AGGREGATED GROUP',5X,'LABOR',5X,'NOBS',/,
     126X,'RATE',/)
      NCARD=0
      LL=1
      TMHR=0.0
      TCOST=0.0
C
C   READ INPUT DATA BY AGGREGATION GROUPS AND COMPUTE MEANS
C
20    READ (5,9000,END=999) N
9000  FORMAT (I3)
      NCARD=NCARD+N
      SUM=0.0
      TOT=0.0
      DO 1000  I=1,N
      READ (5,9001) IB,AB,KB,DMHR,DLBC
9001  FORMAT (I2,A8,A3,22X,F6.0,F7.0)
      SUM=SUM+DLBC
      TOT=TOT+DMHR
1000  CONTINUE
      TMHR=TMHR+TOT
      TCOST=TCOST+SUM
      DOLRT=SUM/TOT
C
C   WRITE MEAN LABOR COST FOR EACH GROUP AGGREGATED
C
      WRITE (6,9501)LL,IB,AB,KB,DOLRT,N
9501  FORMAT (' ',I6,2X,I2,A8,A3,F10.4,I8)
      LL=LL+1
      IF (LL.EQ.71) WRITE (6,9400)
      GO TO 20
C
C   WRITE WEIGHTED AVERAGE FOR ALL GROUPS
C
999   DOLRT=TCOST/TMHR
      WRITE (6,9502) DOLRT,NCARD
9502  FORMAT (' ',//, 5X,'DOLLARS PER LABOR HOUR =',F11.5,/,
     1 5X,'WEIGHTED AVERAGE OF THE ABOVE LABOR RATES',//,
     2 5X,'TOTAL NO. OF OBSERVATIONS =',I6)
      STOP
      END
```

APPENDIX K

MPS/360 Data Preparation Program

A. Purpose: This program was prepared to assist the user in preparing input data decks for use with MPS/360. The required input formats are given in Chapter 4 of Ref. 7.

B. Inputs: A special sequence of input data cards is required. The input data cards must be in the order specified. The number of data cards in each of the below sections is not important because the program will read cards sequentially. For example, if a particular format for data requires a 10 number field with 5 columns for each number and the input data is 15 numbers then the program will read 10 numbers from the first input card and 5 numbers from the second one. All remaining columns of the second card should be blank.

(1) The model name is punched in CC 1-8. It must be alphabetic and not begin with "X". It must be left adjusted in the field. If right hand side (RHS) parametric study vectors are required then put a "1" in CC 9. The RHS parametric vectors are "change columns" and are described in Ref. 7. They consist of a RHS column vector with zeros in every row except a "-1" in one of the resource rows. Three separate decks are created with one corresponding to each resource row. These decks

are used with MPS/360 to provide the parametric studies on the resource vector R.

(2) The number of rows in the T_1 matrix is punched in CC 1-3. The number of columns of T_1 is punched in CC 4-6.

(3) The penalty costs must be punched in this set of cards. Each card may have only 10 numbers with a 5 column field width. The computer program automatically places a decimal after the first three digits in any field. For example, if 12282 is in the 5 column field then the penalty cost is $122.82. There should be one penalty cost for each column of T_1.

(4) Each column of T_1 must be assigned a reference number for MPS/360. Begin with "1" and continue to number columns until all are numbered. These numbers must be punched 40 numbers per card with a 2 column field width. <u>They must be left adjusted</u>.

(5) Each column of T_1 produces some particular type of output. To associate each column with an output a set of cards punched with "T_1 ROW" as described in Chapter IV must be provided. Use the same format as in (4).

(6) The production vector (in man-hours) must be punched in this set of cards. The proper format is 10 numbers per card with a 5 column field width. There should be as many numbers as rows of T_1. Use only integer values.

(7) The resource vector, R, is specified in this card. Use a 10 column field width for the three values of R1, R2 and R3 respectively. Use only integer values.

(8) The last input is the deck of aggregated data cards whose format is specified in Appendix F. They must be in the proper order to correspond with (3) and (5) above.

C. Outputs: A deck of cards is punched for use with MPS/360. However, the user must provide the MPS/360 control deck as described in Refs. 6 and 7.

D. Special Notes:

(1) Be sure not to exceed the field limits specified above.

(2) The maximum number of rows and columns for the T_1 matrix is 99 in this program. See (2) above. This also limits other inputs to a maximum of 99 input numbers. Change DIMENSION and FORMAT statements if further capacity is desired.

```
C*****************************************************************************
C
C
C    CALCULATION AND PUNCHING OF INPUT DATA DECK FOR MPS/360
C
C
C*****************************************************************************
      DIMENSION OBJ(100),C(100),X(5,100),IB(100),JB(100),KB(100),NB(100)
      DIMENSION M(100),L(100),RHS(100),R(3),AB(2)
      INTEGER*2 M,L
      REAL*8 JB
      DATA ONE/-1.0/
C
C    READ INPUT DATA
C
2     READ (5,4,END=9999) (AB(I),I=1,2),MPARA
4     FORMAT (2A4,I1)
      READ (5,5) P
5     FORMAT (F3.2)
      READ (5,7) NROW,NCOL
7     FORMAT(2I3)
      READ (5,8) (C(I),I=1,NCOL)
8     FORMAT (10F5.2)
      READ (5,9) (M(I),I=1,NCOL)
9     FORMAT (40A2)
      READ (5,10) (L(I),I=1,NCOL)
10    FORMAT (40A2)
      READ (5,11) (RHS(I),I=1,NROW)
11    FORMAT (10F5.0)
      READ (5,12) (R(I),I=1,3)
12    FORMAT (3F10.0)
      READ (5,17)((IB(I),JB(I),KB(I),(X(J,I),J=1,5),NB(I)),I=1,NCOL)
17    FORMAT (A2,A8,A3,2F9.2,9X,2F9.2,9X,F9.2,I4)
C
C    WRITE INPUT DATA
C
      WRITE (6,95)
95    FORMAT ('1',6X,'IDENTIFICATION',8X,'NORM',11X,'DMHR',11X,'DML$',
     111X,'DOH$',11X,'NDAYS',7X,'NOBS',//)
      DO 100 I=1,NCOL
      IF (I.EQ.71) WRITE (6,95)
      WRITE (6,97) I,IB(I),JB(I),KB(I),(X(J,I),J=1,5),NB(I)
97    FORMAT(' ',I4,2X,A2,A8,A3,5F15.3,I8)
100   CONTINUE
C
C    NORMALIZE ACTIVITY VECTORS
C
      DO 200 I=1,NCOL
```

```
      Z=X(1,I)
      DO 150 J=1,5
150   X(J,I)=X(J,I)/Z
200   CONTINUE
C
C    CALCULATE AND PRINT ALL LP VALUES
C
      WRITE (6,250) (AB(I),I=1,2)
250   FORMAT ('1',15X,'MODEL NAME = ',2A4,//)
      WRITE (6,260)
260   FORMAT (' ',10X,70HCOL NO.   ROW NO.   OBJ ROW   E ROW     R1 ROW
     1   R2 ROW    R3 ROW    ,//)
      DO 300 I=1,NCOL
      IF (I.EQ.71) WRITE (6,1001)
1001  FORMAT ('1',10X,70HCOL NO.   ROW NO.   OBJ ROW   E ROW     R1 ROW
     1   R2 ROW    R3 ROW    ,//)
      OBJ(I)=P*X(2,I)+X(3,I)+X(4,I)+C(I)*X(5,I)
      WRITE (6,301) L(I),M(I),OBJ(I),ONE,(X(J,I),J=2,4)
301   FORMAT (' ',11X,A2,8X,A2,3X,5F10.4)
300   CONTINUE
      WRITE (6,305)
305   FORMAT ('1',15X,'RIGHT HAND SIDE VALUES',///,15X,'ROW NO.',
     19X,'RHS VALUE',//)
      WRITE (6,306) ((L(I),RHS(I)),I=1,NROW)
306   FORMAT (' ',16X,'E',A2,6X,F12.2)
      WRITE (6,307) ((I,R(I)),I=1,3)
307   FORMAT (' ',16X,'R',I1,7X,F12.2)
C
C    PUNCH INPUT DECK FOR MPS/360
C
      WRITE (7,310)
310   FORMAT ('//MPS2.SYSIN  DD  *')
      WRITE (7,320) (AB(I),I=1,2)
320   FORMAT ('NAME',10X,2A4)
      WRITE (7,350)
350   FORMAT ('ROWS',/,1X,'N',2X,'OBJ')
      DO 400 I=1,NROW
      WRITE (7,401) L(I)
401   FORMAT (1X,'L',2X,'E',A2)
400   CONTINUE
      DO 500 I=1,3
      WRITE (7,501) I
501   FORMAT (1X,'L',2X,'R',I1)
500   CONTINUE
      WRITE (7,550)
550   FORMAT ('COLUMNS')
      DO 600 I=1,NCOL
      WRITE (7,601) L(I),OBJ(I)
```

```
601   FORMAT (4X,'Z',A2,7X,'OBJ',7X,F12.4)
      WRITE (7,602) L(I),M(I),ONE,X(2,I)
602   FORMAT (4X,'Z',A2,7X,'E',A2,7X,F12.4,3X,'R1',8X,F12.4)
      WRITE (7,603) L(I),X(3,I),X(4,I)
603   FORMAT (4X,'Z',A2,7X,'R2',8X,F12.4,3X,'R3',8X,F12.4)
600   CONTINUE
      WRITE (7,750)
750   FORMAT ('RHS')
      DO 800 I=1,NROW
      RHS(I)=-RHS(I)
      WRITE (7,801) L(I),RHS(I)
801   FORMAT (4X,'RHS',7X,'E',A2,7X,F12.4)
800   CONTINUE
      DO 900 I=1,3
      WRITE (7,901) I,R(I)
901   FORMAT (4X,'RHS',7X,'R',I1,8X,F12.4)
900   CONTINUE
      IF (MPARA.EQ.0) GO TO 945
      DO 910 J=1,3
      DO 920 I=1,NROW
      WRITE (7,921) J,L(I)
921   FORMAT (4X,'RHS',I1,6X,'E',A2,13X,'0.0')
920   CONTINUE
      DO 930 K=1,3
      IF (J.EQ.K) GO TO 940
      WRITE (7,931) J,K
931   FORMAT (4X,'RHS',I1,6X,'R',I1,14X,'0.0')
      GO TO 930
940   WRITE (7,941) J,K
941   FORMAT (4X,'RHS',I1,6X,'R',I1,13X,'-1.0')
930   CONTINUE
910   CONTINUE
945   WRITE (7,950)
950   FORMAT ('ENDATA')
      GO TO 2
9999  WRITE (6,1000)
1000  FORMAT ('1')
      STOP
      END
```

APPENDIX L

Cluster Centroid Dominance Program

A. Purpose: This program takes an input deck of aggregated data cards (processes) and a set (more than one if desired) of cluster descriptions and produces a dominance matrix showing the relationships between cluster centroids (or "aggregated" processes).

B. Inputs: A control card with the number of aggregated data cards punched in CC 1-3 is first. Aggregated data cards then follow this card. Next the user may put several cluster "descriptions" if desired. (At least one is required.) The cluster "description" format is specified in Appendix E.

C. Outputs: A cluster description, centroid data listing and dominance matrix are provided for each cluster description. A list of efficient groups is listed. An efficient group is a centroid that is not dominated by any other centroid.

D. Special Notes:

(1) The dominance matrix shows an "O" where each centroid is either dominates or is dominated. The number of "O's" in each row and column are listed at the sides of the matrix. A "-" means no dominance relationship exists.

(2) The maximum number of centroids in any cluster description is 50.

(3) The maximum number of aggregated data cards is 100.

```
C***********************************************************************
C
C
C        CHECK CLUSTER CENTROIDS FOR DOMINANCE
C           THIS PROGRAM WILL TAKE MULTIPLE CLUSTER
C           DESCRIPTIONS FOR ANALYSIS
C
C
C***********************************************************************
      DIMENSION X(6,100),MCLUS(100),LOCN(101),NAME(100),NUMB(100),
     1  CENTR(6,50),D(5),IOUT(50,50),IEFF(50),ITOT(50),JTOT(50),Y(6)
      DATA IDASH/'-'/,IOH/'O'/,ISTR/'*'/
      DATA Y/'NORM','DMHR','DML$','DOH$','NDAY','NOBS'/
      ISW=0
      READ(5,7) N
7     FORMAT(I3)
C
C   READ INPUT DATA
C
      READ(5,17)((X(J,I),J=1,6),I=1,N)
17    FORMAT (13X,2F9.2,9X,2F9.2,9X,F9.2,F4.0)
C
C   READ CLUSTER DESCRIPTIONS
C
20    READ (5,7,END=9999) NCLUS
      READ(5,37)(NUMB(I),I=1,NCLUS)
37    FORMAT(20I3,20X)
      READ(5,37)(MCLUS(I),I=1,N)
      LOCN(1)=1
      DO 50 K=1,NCLUS
      ITOT(K)=0
      JTOT(K)=0
      LOCN(K+1)=LOCN(K)+NUMB(K)
      NAME(K)=MCLUS(LOCN(K))
      DO 50 KK=1,NCLUS
      IOUT(K,KK)=IDASH
50    CONTINUE
      DO 60 K=1,NCLUS
60    IOUT(K,K)=ISTR
      NEFF=0
C
C   WRITE INPUT DATA
C
      IF(ISW.EQ.1) GO TO 70
      WRITE (6,62) Y
62    FORMAT ('1',1X,5A15,A13,/)
      DO 99 I=1,N
      WRITE (6,97) I,(X(J,I),J=1,6)
```

```
97    FORMAT (' ',I4,5F15.5,F8.0)
      IF (I.EQ.70) WRITE (6,62) Y
99    CONTINUE
C
C   WRITE CLUSTER
C
70    WRITE (6,65)
65    FORMAT ('1',//)
      WRITE(6,287) NCLUS
287   FORMAT('OSTAGE',I4)
      DO 290 I=1,NCLUS
      L=LOCN(I)
      LL=LOCN(I+1)-1
290   WRITE(6,297)I,NAME(I),(MCLUS(K),K=L,LL)
297   FORMAT(I5,' GROUP',I4,I10,19I4,/,(16X,20I4))
C
C   COMPUTE WEIGHTED GROUP CENTROIDS
C
      DO 100 K=1,NCLUS
      DO 100 J=1,5
100   CENTR(J,K)=0.0
      DO 150 K=1,NCLUS
      ZNK=0.0
      NK=NUMB(K)
      L=LOCN(K)-1
      DO 140 I=1,NK
      MC=MCLUS(L+I)
      ZNK=ZNK+X(6,MC)
      DO 140 J=1,5
140   CENTR(J,K)=CENTR(J,K)+X(J,MC)*X(6,MC)
      DO 145 J=1,5
145   CENTR(J,K)=CENTR(J,K)/ZNK
      CENTR(6,K)=ZNK
150   CONTINUE
C
C   WRITE WEIGHTED GROUP CENTROIDS
C
      IF (NCLUS.GT.30) WRITE (6,65)
      WRITE(6,147) Y
147   FORMAT (' ',///,21X,5A15,A13,/)
      WRITE(6,155)(K,NAME(K),(CENTR(J,K),J=1,6),K=1,NCLUS)
155   FORMAT(I5,' GROUP',I4,' CENTROID',5F15.5,F8.0)
C
C   NORMALIZE CENTROIDS TO UNIT OUTPUT
C
      DO 200 K=1,NCLUS
      Z=CENTR(1,K)
      DO 200 J=1,5
```

```
200   CENTR(J,K)=CENTR(J,K)/Z
C
C    WRITE NORMALIZED CENTROIDS
C
      WRITE (6,148) (Y(I),I=1,5)
148   FORMAT ('1',///,23X,5A10,/)
      WRITE(6,157)(K,NAME(K),(CENTR(J,K),J=1,5),K=1,NCLUS)
157   FORMAT(I5,' GROUP',I4,' CENTROID',5F10.3)
C
C    CHECK FOR DOMINANCE AND STORE RELATIONSHIPS
C
      NCLUM=NCLUS-1
      DO 400 K=1,NCLUM
      KP1=K+1
      DO 400 KK=KP1,NCLUS
      DO 250 J=2,5
250   D(J)=CENTR(J,K)-CENTR(J,KK)
      IF((D(2).GE.0).AND.(D(3).GE.0).AND.(D(4).GE.0).AND.(D(5).GE.0))
     1  GO TO 300
      IF((D(2).LE.0).AND.(D(3).LE.0).AND.(D(4).LE.0).AND.(D(5).LE.0))
     1  GO TO 350
      GO TO 400
300   IOUT(K,KK)=IOH
      ITOT(K)=ITOT(K)+1
      JTOT(KK)=JTOT(KK)+1
      GO TO 400
350   IOUT(KK,K)=IOH
      ITOT(KK)=ITOT(KK)+1
      JTOT(K)=JTOT(K)+1
400   CONTINUE
C
C    WRITE DOMINANCE MATRIX
C    ( IF NUMBER OF CLUSTERS IS 30 OR LESS )
C
      IF (NCLUS.GT.30) GO TO 600
      WRITE (6,149)
149   FORMAT(' ',//)
      WRITE (6,502) (I,I=1,NCLUS)
502   FORMAT (' ',5X,'GROUP',5X,'DOM BY',3X,50I3)
      WRITE (6,398)
398   FORMAT (' ')
      DO 500 I=1,NCLUS
      WRITE (6,501) I,ITOT(I),(IOUT(I,J),J=1,NCLUS)
501   FORMAT (' ',7X,I3,6X,I4,6X,50A3)
      IF (ITOT(I).NE.0) GO TO 500
      IEFF(NEFF+1)=I
      NEFF=NEFF+1
500   CONTINUE
```

```
      WRITE (6,398)
      WRITE (6,554) (JTOT(I),I=1,NCLUS)
554   FORMAT (' ',24X,50I3)
C
C    WRITE THE EFFICIENT GROUPS
C
      WRITE (6,149)
      WRITE (6,555) (IEFF(I),I=1,NEFF)
555   FORMAT (' ',10X,'THE EFFICIENT GROUPS ARE' ,25I3)
      ISW=1
      GO TO 20
C
C    WRITE DOMINANCE MATRIX
C    ( IF NUMBER OF CLUSTERS IS GREATER THAN 30 )
C
600   WRITE (6,65)
      WRITE (6,602) (I,I=1,NCLUS)
602   FORMAT (' ',5X,'GROUP',5X,'DOM BY',4X,50I2)
      WRITE (6,398)
      DO 700 I=1,NCLUS
      WRITE (6,601) I,ITOT(I),(IOUT(I,J),J=1,NCLUS)
601   FORMAT (' ',7X,I3,6X,I4,6X,50A2)
      IF (ITOT(I).NE.0) GO TO 700
      IEFF(NEFF+1)=I
      NEFF=NEFF+1
700   CONTINUE
      WRITE (6,398)
      WRITE (6,654) (JTOT(I),I=1,NCLUS)
654   FORMAT (' ',25X,50I2)
      WRITE (6,149)
      WRITE (6,555) (IEFF(I),I=1,NEFF)
      ISW=1
      GO TO 20
9999  STOP
      END
```

LIST OF REFERENCES

1. Baumol, W.J., Economic Theory and Operations Analysis, 2nd ed., Prentice Hall, 1965.

2. Dorfman, R., Samuelson, P.A., and Solow, R.M., Linear Programming and Economic Analysis, McGraw Hill, 1958.

3. Ferguson, C.E., Microeconomic Theory, 3rd ed., Irwin, 1972.

4. Gale, D., The Theory of Linear Economic Models, McGraw Hill, 1960.

5. Henderson, J.M. and Quandt, R.E., Microeconomic Theory, 2nd ed., McGraw Hill, 1971.

6. International Business Machines Corporation, Mathematical Programming System/360 Version 2, Control Language User's Manual, 4th ed., 1968.

7. International Business Machines Corporation, Mathematical Programming System/360, Linear and Separable Programming User's Manual, 2nd ed., 1968.

8. Lance, G.N. and Williams, W.T., "A General Theory of Classificatory Sorting Strategies, 1. Hierarchical Systems," The Computer Journal, v. 9, No. 4, p. 373-380, February 1967.

9. Mansfield, E., Microeconomics, Norton, 1970.

10. Naval Air Rework Facility Instruction 7650.1A dated 28 April 1971, with Change 1 dated 30 July 1971; Subject: Cost Control Manual.

11. Spooner, R.L., Evaluation of a Technology Change in Production, M.S. Thesis, Naval Postgraduate School, Monterey, 1972.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Asst. Professor N.K. Womer, Code 55Wr Department of Operations Research and Administrative Sciences Naval Postgraduate School Monterey, California 93940	2
4.	Asst. Professor J.K. Hartman, Code 55 Hh Department of Operations Research and Administrative Sciences Naval Postgraduate School Monterey, California 93940	1
5.	Chief of Naval Personnel Pers - 11b Department of the Navy Washington, D.C. 20370	1
6.	Department of Operations Research and Administrative Sciences, Code 55 Naval Postgraduate School Monterey, California 93940	1
7.	LCDR Tom Kennepell, USN MSDO Representative Naval Air Rework Facility Naval Air Station, Alameda Alameda, California 94501	1
8.	Management Systems Development Office Attn: Mr. Pete Baxter Naval Air Station, North Island Coronado, California	2
9.	Dr. T. Varley Office of Naval Research Code 521 Washington, D.C. 20352	1
10.	Lt. William M. Myers, SC, USN c/o Z.T. Myers, Jr. 3101 Goneaway Road Charlotte, North Carolina 28210	2

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)	2a. REPORT SECURITY CLASSIFICATION
Naval Postgraduate School Monterey, California 93940	Unclassified
	2b. GROUP

3. REPORT TITLE

A Linear Economic Model for a Naval Air Rework Facility

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Master's Thesis; September 1972

5. AUTHOR(S) (First name, middle initial, last name)

William Martin Myers

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
September 1972	114	11

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Naval Postgraduate School Monterey, California 93940

13. ABSTRACT

A prototype management information system called Work-in-Process Inventory Control System (WIPICS) has been installed at the Naval Air Rework Facility (NARF), North Island. WIPICS is expected to benefit the NARF by reducing rework times and rework costs. By assuming that linear production processes may be estimated from statistical data accumulated by the NARF, a linear economic model is constructed which predicts a required budget in man-hours, material cost and overhead cost categories for a specified production output level. Sample problems are solved and parametric studies are done to determine changes in the required budget for restrictions in man-hours, material cost and overhead cost. The model will be used as an aid in the cost-effectiveness evaluation of WIPICS by the Management Systems Development Office.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6811

Security Classification

A-31408

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Naval Air Rework Facility						
Work-in-Process Inventory Control System						
Linear Economic Model						
Linear Production Model						
Management Systems Development Office						

138226
Thesis
M997 Myers
c.1 A linear economic
model for a Naval Air
Rework Facility.